AÑO III - NÚM. 76

MICROHOBBY SEMANAL

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

AÑO III - N.º 76

135 PTS.
Canarias 140 ptas.

MICROPANORAMA

AMSTRAD POTENCIARÁ LOS PRODUCTOS SINCLAIR

EXCLUSIVA

PROTAGONISTAS PARA TUS PROGRAMAS

¡DISEÑADOS POR DINAMIC!

MICROFILE

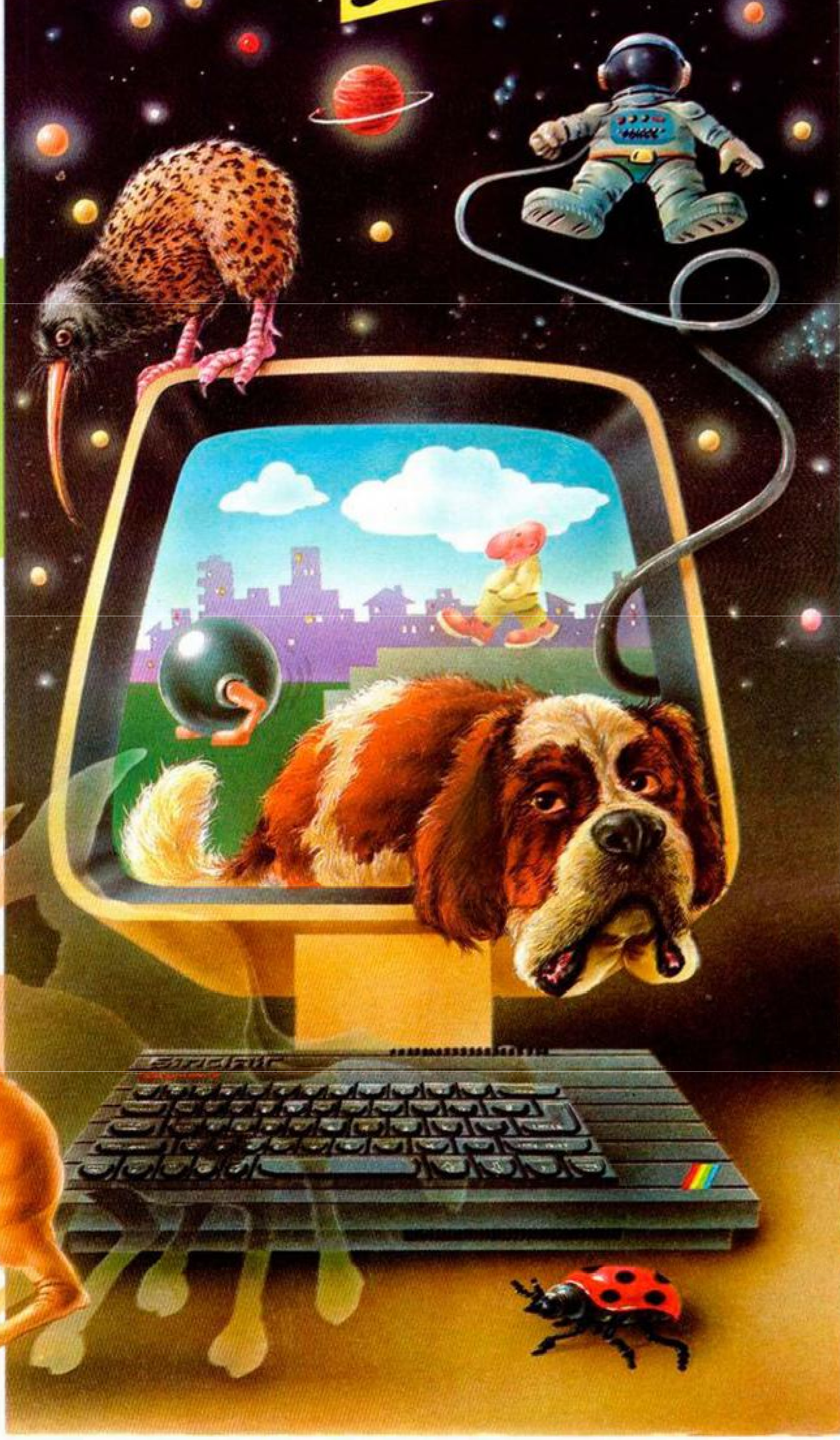
MAPA DE LOS SECTORES DE UN MICRODRIVE

NUEVO

2112 EL IMPERIO DE LOS ORDENADORES

TRUCOS

SIMULACIÓN DE LA SENTENCIA "SOUND"



HOBBY PRESS

Director Editorial
José I. Gómez-Centurión
Director Ejecutivo
Domingo Gómez
Asesor Editorial
Gabriel Nieto
Redactora Jefe
Africa Pérez Tolosa
Diseño
Rosa María Capitel
Redacción
Amalio Gómez, Pedro Pérez, Jesús Alonso
Secretaria Redacción
Carmen Santamaría
Colaboradores
Primitivo de Francisco, Rafael Prades, Miguel Sepúlveda, Sergio Martínez y J. M. Lazo
Corresponsal en Londres
Alan Heap
Fotografía
Carlos Candel
Portada
José María Ponce
Dibujos
Teo Mójica, F. L. Frontán, J. M. López Moreno, J. Igual, J. A. Calvo, Lóriga, J. Olivares
Edita
HOBBY PRESS, S. A.
Presidente
María Andrino
Consejero Delegado
José I. Gómez-Centurión
Jefe de Producción
Carlos Peropadre
Jefe de Publicidad
Concha Gutiérrez
Publicidad Barcelona
José Galán Cortés
Tels.: 303 10 22 - 313 71 76
Secretaria de Dirección
Marisa Cogorro
Suscripciones
M.ª Rosa González
M.ª del Mar Calzada
Redacción, Administración y Publicidad
La Granja, 39
Polígono Industrial de Alcobendas
Tel.: 654 32 11
Telex: 49480 HOPR
Dto. Circulación
Paulino Blanco
Distribución
Coedis, S. A. Valencia, 245
Barcelona
Imprime
Rotedic, S. A. Ctra. de Irún, km 12,450 (MADRID)
Fotocomposición
Novocomp, S.A.
Nicolás Morales, 38-40
Fotomecánica
Grof
Ezequiel Solana, 16
Depósito Legal
M-36.598-1984
Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina)

Solicitado control OJD

MICROHOBBY ESTA SEMANA

AÑO III. N.º 76. 29 de abril al 5 de mayo de 1986.
135 ptas. Canarias, Ceuta y Melilla: 130 ptas.
Sobretasa aérea para Canarias: 10 ptas.





¿Te imaginas un mundo dominado por ordenadores?... (Pág. 12)

MICROHOBBY NUMEROS ATRASADOS

Queremos poner en conocimiento de nuestros lectores que para conseguir números atrasados de MICROHOBBY SEMANAL, no tienen más que escribirnos indicándonos en sus cartas el número deseado y la forma de pago elegida de entre las tres modalidades que explicamos a continuación.
Una vez tramitado esto, recibirá en su casa el número solicitado al precio de 95 ptas. + 6 de IVA hasta el n.º 36, a 125 ptas. + 8 de IVA hasta el n.º 60 y a 135 ptas. desde el n.º 60 en adelante.



FORMAS DE PAGO

- Enviando talón bancario nominativo a Hobby Press, S. A. al apartado de Correos 54062 de Madrid.
- Mediante Giro Postal, indicando número y fecha del mismo.
- Con Tarjeta de Crédito (VISA o MASTER CHARGE), haciendo constar su número y fecha de caducidad.

ALAN SUGAR, DIRECTOR DE AMSTRAD, POTENCIARA LOS ORDENADORES SINCLAIR



Nuestro corresponsal en Londres, Alan Heap, nos informa de las últimas declaraciones de Alan Sugar, «Chairman» de Amstrad, referentes a sus nuevas relaciones con Sinclair Research.

Directamente desde Inglaterra, Sugar ha declarado en un encuentro con la prensa británica, que utilizará su propia red mundial de distribución para la venta de los productos Sinclair.

Recordemos que no hace mucho Amstrad se hizo cargo de los derechos mundiales de todos los productos Sinclair, es decir, distribución y comercialización de las creaciones «relevantes» de la compañía, tanto actuales como futuras.

Sugar ha manifestado con rotundidad, que su política no causará ningún perjuicio en absoluto a los usuarios de Sinclair, y que seguirán contando con todo tipo de asistencia a todos los niveles, tanto técnica como de reparación y postventa en general.

La compra de estos derechos ha supuesto para Amstrad un desembolso de 5.000.000 de libras, que unido a los 2.000.000 que tuvo que pagar por el stock existente en el mercado, han alcanzado la cifra de 7 millones lo que totaliza el capital que Sinclair adeudaba a otras compañías.

En cuanto a las medidas referidas a los modelos de ordenadores que Sinclair posee actualmente entre su gama de productos, parece ser que Amstrad hará algunas importantes transformaciones.

Por ejemplo, en el S 128 K se instalará un cassette como parte integral de la máquina, así como un port para joystick. Este proyecto se llevará a cabo antes de las próximas Navidades y se está estudiando la posibilidad de que el precio del nuevo modelo baje hasta 140 libras, es decir, 40 libras menos que en la actualidad.

Sin embargo, el futuro del QL parece ligeramente más turbio, pues a pesar de ser uno de los mejores micro-ordenadores que existen en el mercado, debido a una mala campaña de marketing, su lanzamiento supuso un fracaso económico para Sinclair, ya que el número de ventas alcanzado en toda Europa fue sensiblemente inferior a lo esperado. A pesar de todo, no se descarta la posibilidad de la fabricación de un nuevo QL, el cual incluirá un disc-drive que vendría a sustituir al micro-drive actual.

En otro orden de cosas, se confirma la noticia de que Sinclair continuará con su misión original de desarrollo de novedades. En este momento se está trabajando sobre dos nuevos proyectos: los silicon-wafers y el micro-ordenador portátil PANDORA, por los cuales Amstrad ha mostrado un gran interés.

Por desgracia, todo tiene su parte negativa, y ésta le ha tocado a los trabajadores de Sinclair, los cuales tendrán que sufrir una considerable reducción de plantilla, sobre todo en lo que corresponde a los departamentos de distribución y marketing.

Por último, señalar que Amstrad ha propuesto que el Spectrum 128 siga siendo fabricado por las compañías que lo venían haciendo hasta el momento: Timex, AB Electronic y Thorn, pues no tienen planeado llevar esta nueva producción a sus fábricas en Corea, donde se realiza en la actualidad toda la gama CPC de Amstrad.

CYBERUN Y PENTAGRAM PRONTO A LA VENTA

Dos nuevos, novísimos, programas de Ultimate harán aparición próximamente en el mercado dispuestos a acaparar la atención y el interés de todos. Se trata de Cyberun y Pentagram, dos excelentes arcades en la más pura línea Jet Pack, el primero y Knight Lore, el segundo.

Cyberun, como algunos de vosotros ya sabréis, consiste en una carrera espacial en la que tendremos que luchar por conseguir la sustancia más poderosa y destructiva del Universo: el Cyberun. Para ello previamente deberemos formar nuestra propia nave, cuyos trozos se encuentran esparcidos por la superficie de un escabroso planeta y posteriormente, luchar contra los numerosísimos contrincantes y meteoritos que irán apareciendo en su largo y duro camino.

Por su parte, **Pentagram** es un nuevo arcade que supone la vuelta de Ultimate a su innovadora y personal Técnica Filmation, la cual fue empleada por primera vez en Knight Lore y posteriormente en Alien 8.

Con esto parece que se ha abandonado, al menos temporalmente, el Filmation II que se había venido utilizando en las últimas creaciones como Nightshade o Gunfright, pues parece ser que la primera de las modalidades ofrece unos resultados mucho más vistosos del mismo modo que aporta un mayor número de diferentes posibilidades de movimiento para el personaje.

La novedad que presenta este Pentagram es que mientras que en sus predecesores podíamos pasar largo tiempo planeando la estrategia a seguir, aquí deberemos reaccionar rápidamente, pues al cabo de unos pocos segundos comenzarán a aparecer en la pantalla los tan temidos enemigos.

Mastercomputer celebró su concurso

UN ROTUNDO EXITO

Hace algunas semanas os informábamos acerca del concurso que Mastercomputer había organizado en torno al excelente programa de Melbourne, GYROSCOPE.

Este concurso se llevó a cabo tal y como estaba previsto y su organización supuso un rotundo éxito.

El ganador de tan trepidante concurso fue Jaime Borrell, quien joystick en ristre demostró ser poseedor de unos nervios de acero y de un pulso sin igual. Sin embargo, la batalla fue dura, pues todos los participantes lucharon denodadamente hasta el último segundo en pos de la victoria.

El resultado fue que todos se lo pasaron en grande (como muy bien se puede comprobar en sus caras) y que al final hubo premios y regalos para todos. Esperamos que sigan organizándose concursos de este tipo en los que los chavales puedan pasar buenos ratos gracias a la diversión que los ordenadores les pueden ofrecer.



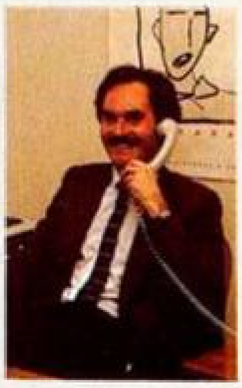
EL CONCURSO MASTERMIND YA TIENE GANADORES

El Concurso MasterMind, patrocinado conjuntamente por MicroWorld y Microhobby, ya tiene ganadores.
Tras la celebración de la cuarta y última fase del concurso, el pasado 18 de marzo, y las eliminatorias posteriores llevadas a cabo entre los finalistas el 1 de abril, éstos han sido los galardonados:

1. D. Fernando Rada (N.° 229). Ganador de un viaje a Londres para 2 personas.
2. D. Francisco Soto Espinosa (N.° 18). Ganador de un Monitor Color.
3. D. Rafael de las Heras (N.° 54). Ganador de un Spectrum Plus.
4. D. Juan José Ibáñez (N.° 287). Ganador de un lote de programas MICROPARADISE y DINAMIC.
5. D. Urbano García Barros (N.° 254). Ganador de un lote de programas MICROPARADISE y DINAMIC.
6. D. Gonzo Suárez (N.° 203). Ganador de un lote de programas MICROPARADISE y DINAMIC.
7. D. Javier Delgado (N.° 149). Ganador de un lote de programas MICROPARADISE y DINAMIC.
8. D. Carlos Granados (N.° 233). Ganador de un lote de programas MICROPARADISE y DINAMIC.
9. D. Francisco Mozo Villapun (N.° 82). Ganador de un lote de programas MICROPARADISE y DINAMIC.
10. D. Alberto Martín (N.° 224). Ganador de un lote de programas MICROPARADISE y DINAMIC.

La entrega de premios se efectuará el día 7/V/1986 a las 19,00 horas en la tienda de MICROWORLD, S. A. de la calle José Ortega y Gasset, 21 (Madrid).

AQUI LONDRES



Muchas compañías de software se muestran preocupadas por el retraso tras la publicidad inicial de un juego y la aparición del mismo en el mercado, pero es cierto que, en la mayoría de los casos, la culpable es la propia casa por anunciar su producto antes de que éste esté finalizado, lo que produce que, en el caso de que existan problemas en el desarrollo del juego, el retraso, en algunos casos, sea demasiado extenso. La solución, está claramente en sus manos.

Off the Hook, la cinta recopilación a beneficio de la guerra contra la droga, ya está a la venta en sus versiones para **Spectrum** y **C-64,** y a un precio de 7 libras. Próximamente estarán disponibles las versiones para Amstrad y Atari.

El próximo juego de **Melbourne House** será lanzado a finales de abril. Se trata de una aventura gráfica, llamada **Red Hawck.** Melbourne planea lanzarlo simultáneamente para el Spectrum, Commodore y Amstrad, al precio de 9 libras.



Los últimos lanzamientos de software, incluyen títulos como: **Fourmost Adventure** de Global Soft., una cinta recopilación que contiene cuatro aventuras gráficas y que supone una excelente compra para aquéllos a quienes les guste este tipo de juegos. **Devil's Crown** en su versión para Spectrum, ha sido lanzado al precio de 2 libras por Mastertronics, el rey del software barato.

Thirty Games, de Argus Press, es una cinta de recopilación, al precio de 10 libras, lo cual resulta bastante económico si tenemos en cuenta que incluye 30 juegos diferentes.

De nuestro corresponsal en Londres
ALAN HEAP

BOLARINTO

Juanjo BARRERA

Spectrum 48 K

TODAS LAS LETRAS MAYUSCULAS SUBRAYADAS DEBERAN TECLEARSE EN MODO GRAFICO.

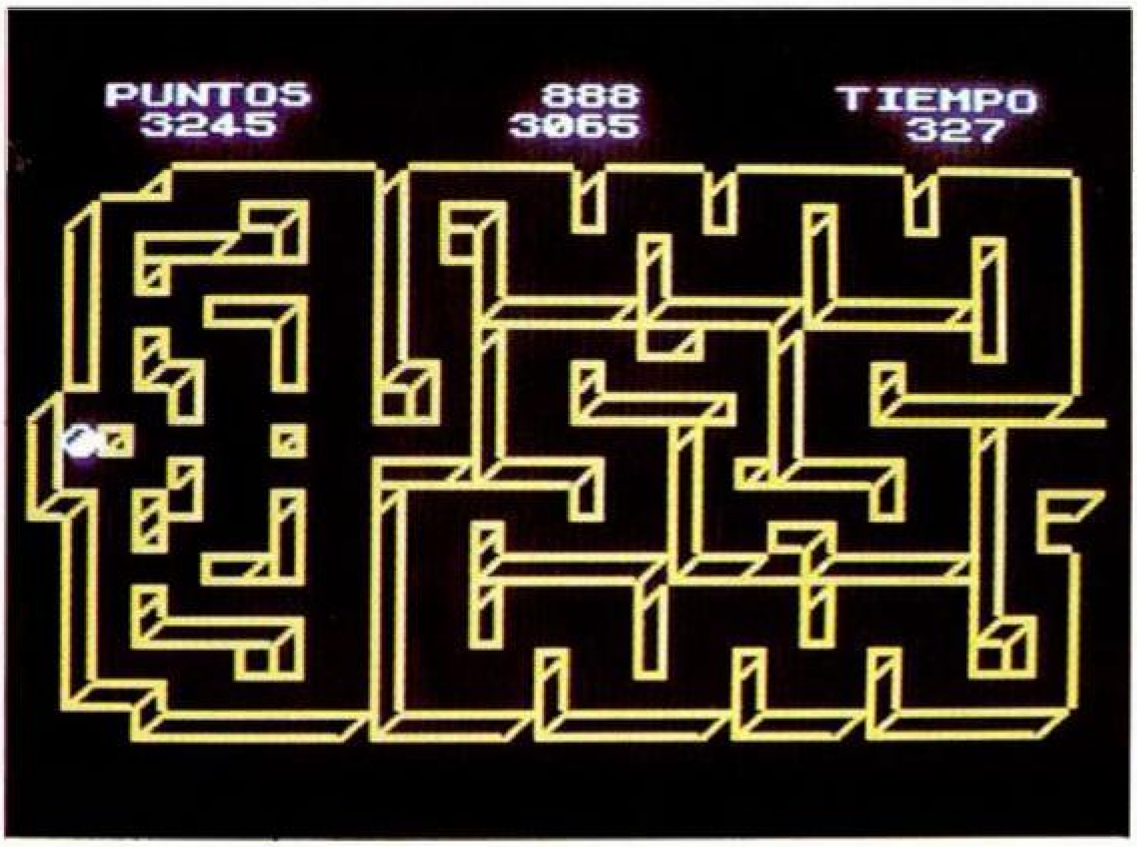
¿Te gustan los juegos de reflejos? Pues con este «Bolarinto» te vas a poner las botas y vas a tener que demostrar todo lo que vales.

El reto a tu habilidad te lo plantea una bola a la que hay que conducir a lo largo de un laberinto, evitando que se caiga por unos agujeros laterales con lo que tendríamos que iniciar de nuevo el recorrido.

En el juego el tiempo es un factor importante, así como conseguir la mayor puntuación posible a lo largo de las cuatro pantallas que lo componen y que van aumentando su dificultad progresivamente.

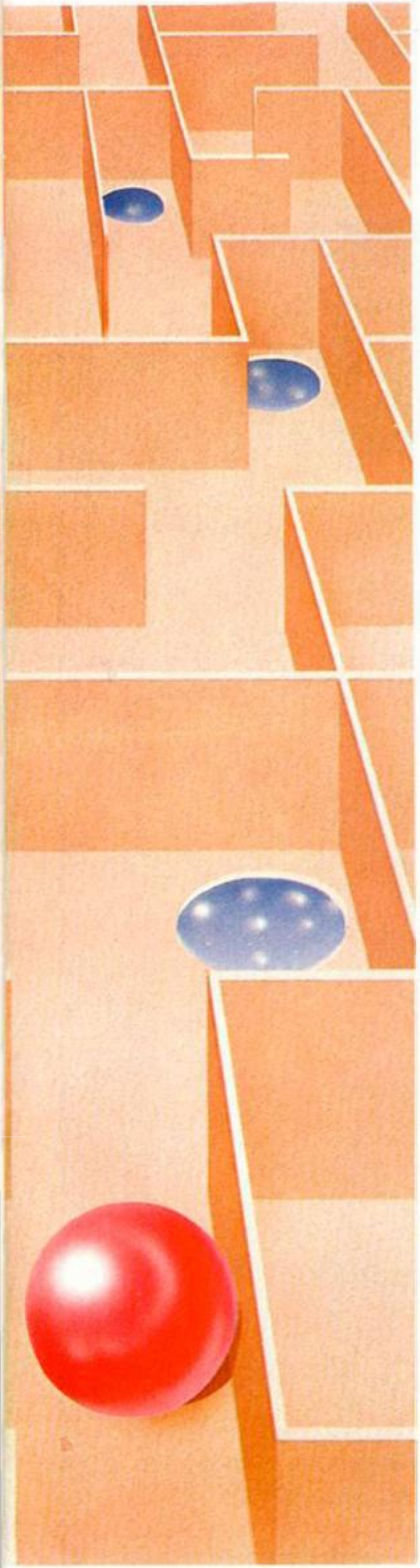
El manejo de las teclas del cursor es el siguiente:

5, a la izquierda
6, hacia abajo
7, hacia arriba
8, a la derecha




```
   1 GO TO 8000
   2 POKE 23606,0: POKE 23607,60
: BORDER 7: PAPER 7: BRIGHT 0: C
LS : INK 0: STOP
  10 PRINT AT c,d;" ";AT a,b;"A"
: LET c=a: LET d=b: LET e=PEEK 2
3560: LET a=a+(e=54)-(e=55): LET
 b=b+(e=56)-(e=53)
  15 IF SCREEN$ (a,b)<>" " AND A
TTR (a,b)<>71 THEN LET a=c: LET
b=d: LET t=t-5: FOR x=10 TO -10
STEP -1: BEEP .02,x: NEXT x: POK
E 23560,0
  20 LET p=p+5: LET t=t-1: PRINT
 AT 1,7-LEN STR$ p;p;AT 1,28-LEN
 STR$ t;" ";t: IF t<=0 THEN PRIN
T #1;"LO SIENTO,SE TERMINO EL TI
EMPO": FOR x=0 TO 200: NEXT x: G
O TO 9500-(2500*(p>hi))
  30 GO TO 10+(6000*(b=31))
1000 GO SUB 5000: LET o=o+25-(25
*(o=50)): LET t=t+125-o
1010 INK 2: LET a$="LOOOOOOOOOOO
OOO": LET b$="RDDDDDDDDDDDDDDH":
FOR x=6 TO 22 STEP 8: FOR y=1 TO
 15: PRINT AT 5+y,x;a$(y);AT 2+y
,x+4;b$(y): NEXT y: NEXT x
1020 PRINT AT 11,26;"B";AT 10,26
;"O";AT 2,3;"PPPPPPP PPPPPPP PPP
PPPP";AT 21,6;"GBBBBBBFGBBBBBBFG
BBBBBBF";AT 10,1;"PH";AT 14,0;"G
BBBBF";AT 10,27;"PPPPP";AT 14,30
;"JF"
1030 LET b$="EDDDDDD": LET d$="E
DDG    ": FOR x=1 TO 7: PRINT AT
2+x,2;b$(x);AT 10+x,0;d$(x);AT 1
4+x,30;"Q";AT 2+x,26;"Q": NEXT x
1040 PRINT AT 21,30;" ": INK 7:
GO TO 10
2010 GO SUB 5000: LET t=t+150-o:
 INK 4: PRINT AT 17,11;"JH";AT 2
,1;"PPPPPPPPPPPPPP PPPPPPP PPPPP
PP";AT 6,4;"JBBBBBH";AT 7,10;"GB
BBBBBBQGBBBBBBBQH";AT 10,1;"BBBBE
I";AT 10,17;"JBBBBBBBBBBBBBQP";AT
 14,1;"BBBBBBBBBQH";AT 17,4;"JH";
AT 17,8;"JH";AT 18,4;"CH";AT 18,
8;"CH";AT 18,11;"CGBBBBGGBBBBI";A
T 14,21;"JBBI";AT 14,27;"JBBBF";
AT 17,25;"JBBH";AT 18,25;"CPPH";
AT 21,0;"GBBBBBBBBBBBBBBBBBBBBBBB
BBBBBBBF"
2020 RESTORE 1020: FOR w=0 TO 10
: READ y: READ z: READ a$: FOR x
=1 TO LEN a$: PRINT AT y+x-1,z;a
$(x): NEXT x: NEXT w
2030 DATA 3,0,"EDDDDDDGEDDGEDDDD
D",8,10,"RDDDDD",10,13,"JOOOOC",1
0,14,"HOOOM",11,17,"RDDDDDD",3,3
1,"OOOOOOO",15,31,"OOOOOOO",3,15,
"AM",3,23,"AM",5,19,"LD",5,27,"L
D"
```





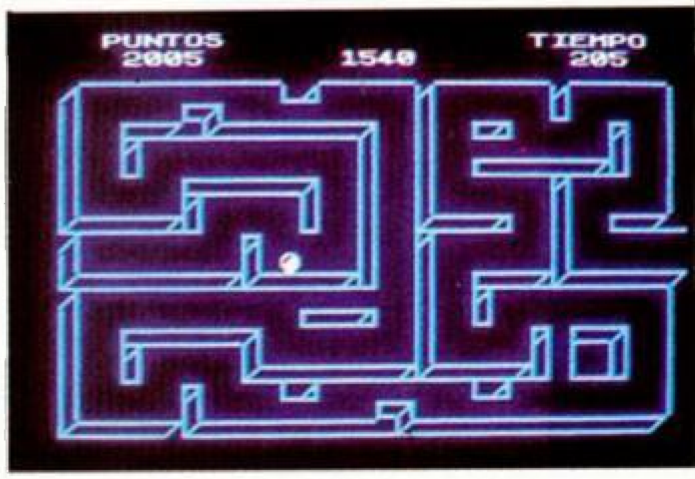



```
2040 INK 7: GO TO 10
3000 GO SUB 5000: LET t=t+225-o:
 INK 5: PRINT AT 2,0;" PPPPPPPPPP
P PPPPPP PPPPPPP PPPPPP E
 CK      R       R     OD    JH
         C       H     OD JBQPGBB
BBBBBH   C  JI      L  OD R
      C  C          D  OD H
      C  C  JBBBBBQR   OD    JBBB
BBH   C  C       R     OD    R
  C   C  C       C     QGBBBBBQR
  H   C  GBBBI   C   JBQPE       L
3010 PRINT                        "
D           D     D     D
GBBBBBBBBBQGBBBBQH     C   JBBQGBBBBBF
E                 C   R           C
C          JBBI   C   H           C
C  JBBBBBH            C     L JBH C
C  R      D           C     D C C C
C  H      GBBBBBBBBBQGBBBBBQR CPH C
C      L     CK          CK       C
C      D           JH             C
GBBBBFGBBBBBBBBBBBPGBBBBBBBBBBBBBBBF"
3030 INK 7: GO TO 10
4000 GO SUB 5000: LET t=t+325-o:
 LET a=a-1: LET c=a: INK 6: PRIN
T AT 2,0;"      PPPPPPP PPPPPP P
PP PPPPPP PPPP    PK       R       R
  R       R    O   E   JH   C  JH  H
  H   L   H    C   C JBQPH  C  CC
L     C      L C   C I      C  C
C     C      C C   C    JBH C  GBBBQ
GBBBFGBBBBQC   C   C      C C  C
CK  R        H C   H GH   H GH C  L
    C    L     QE     H     CH C  GB
BBI C    GBBBBBQPC  S     S    C
  R C        R   C   L       JBQGBBH
  C UGBBH    C
4010 PRINT                        "
GH LN   L    R        H D     H D JE
 C  I   C    C     L    C JH    C C
 C    JQR    C     GBBBBBFGBQPGBBBBQC C
 C  L        C     R     R    R   C  C
 C  GBBBH    C     H     C    H   GH C
 C      CH   C         L H L    L CH C
 GBH         C         D   D    D    C
   GBBBBBBFGBBBBFGBBBBBFGBBFGBBBBBF"
4020 INK 7: GO TO 10
5000 LET a=12: LET c=a: LET b=1
 LET d=b: LET v=t: FOR x=t TO 1
STEP -1: LET p=p+10: LET v=v-1
PRINT AT 1,7-LEN STR$ p;p;AT 1,2
8-LEN STR$ v;" ";v: BEEP .01,5:
NEXT x: FOR x=0 TO 100: NEXT x
5010 CLS : PRINT AT 0,2;"PUNTOS"
;AT 0,24;"TIEMPO";AT 1,7-LEN STR
$ p;p;AT 0,11;x$;AT 1,14;hi: POK
E 23560,0: RETURN
5999 STOP
6010 PRINT AT a,30;" A": LET u=u
+1-(4*(u=4)): GO TO u*1000
7000 CLS : INPUT AT 22,0;AT 10,5
;"HA SUPERADO EL RECORD
      POR ";(p-hi);" PUNTOS

          INTRODUZCA SU NOMBRE

                "; BRIGHT 1; LINE x
$

7010 FOR x=0 TO 10: IF LEN x$<10
 THEN LET x$=" "+x$+" "
7015 NEXT x: LET x$=x$(1 TO 10)
7020 LET hi=p: GO TO 9500
7999
8000 RESTORE 8020: CLEAR 64568:
FOR x=65338 TO 65365: READ y: PO
KE x,y: NEXT x
8010 RANDOMIZE USR 65338
8020 DATA 33,0,61,17,57,252,1,0,
```




```
3,126,203,47,182,18,19,35,11,121
,176,32,244,33,57,251,34,54,92,2
01
8030 LET x$="J.BARREDA ": LET hi
=1000: PRINT AT 11,6;"ESPERE 20
SEGUNDOS"
8997
8998 REM GRAFICOS
8999
9000 DEF FN x(a$,z)=16*(CODE a$(
z)-48-7*(a$(z)>"9"))+(CODE a$(z+
1)-48-7*(a$(z+1)>"9"))
9010 LET a$="3C66CF9FBFFF7E3CFFF
F00000000FFFFC0C0C0C0C0C0FFFFC3C
3C3C3C3C3C3C303070F1B3363C383FFF
F060C183060C0C1C3C6CCD8F0FFFFFFF
F070F1B3363C3FFFF1B3363C3FFFF"
9020 LET a$=a$+"FFFFC0C0C0C0FFFF
03070F1B3363FFFFFFFFC3C7CFDBF3E3
C3C3C3C3C3C3FFFF030303030303FFFF
C0C0C0C0C0C0C0C0000000000000FFFF
FFFF060C1830FFFFC3C7CFDBF3E3C3C3
"
9025 LET a$=a$+"FFFFC7CFDBF3FFFF
C3C7CFDBF3E3FFFFFFFFC3C6CCD8FFFF
"
9030 FOR z=1 TO 336 STEP 2: POKE
 USR "a"+((z/2)-1),FN x(a$,z): N
EXT z
9500 LET c=12: LET d=2: LET a=12
: LET b=2: LET p=0: LET t=150: L
ET o=0: INK 7: BRIGHT 1: PAPER 0
: BORDER 0: CLS : PRINT AT 10,2;
"PULSE UNA TECLA PARA EMPEZAR":
PAUSE 0: CLS : PRINT AT 0,2;"PUN
TOS";AT 0,24;"TIEMPO"
9510 PRINT AT 0,11;x$;AT 1,14;hi
: POKE 23560,0: LET v=1: GO TO 1
010
```

ELIPSES

Este es una subrutina general de construcción de elipses, que nos envía Rafael García González desde Asturias.

Empieza con la línea 9300 y utiliza cuatro variables:

PX: Debe almacenar la coordenada horizontal del centro de la elipse.
PY: Coordenada vertical del centro.
RX: Radio horizontal.
RY: Radio vertical.

```
  10 INPUT px,py,rx,ry
  20 GO SUB 9300
  30 GO TO 10
  40 PRINT
  50 PRINT
9300 FOR n=0 TO 2*PI STEP .1
9310 PLOT px+rx*SIN n,py+ry*COS
n
9320 DRAW px+rx*SIN (n+.1)-PEEK
23677,py+ry*COS (n+.1)-PEEK 2367
8
9340 NEXT n: RETURN
```

SIMULACION DE LA FUNCION «SOUND»

La función «SOUND» de algunos ordenadores la podéis simular fácilmente mediante el programa 1 que nos ha facilitado Juan José Espalza, permitiendo la creación de sonidos mediante los valores del 1 al 255 almacenados en la memoria RAM o en la ROM.

Listado ensamblador

```
10         ORG  32768        100        PUSH HL
20         LD   HL,(#5C76)   110        LD   H,#00
30         PUSH HL           120        LD   L,A
40 BUCLE   LD   D,#00        130        CP   #00
50         LD   A,(#5CB0)    140        JR   Z,#801A
60         LD   E,A          150        CALL #03B5
70         POP  HL           160        JR   BUCLE
80         LD   A,(HL)       170        POP  BC
90         INC  HL           180        RET
```

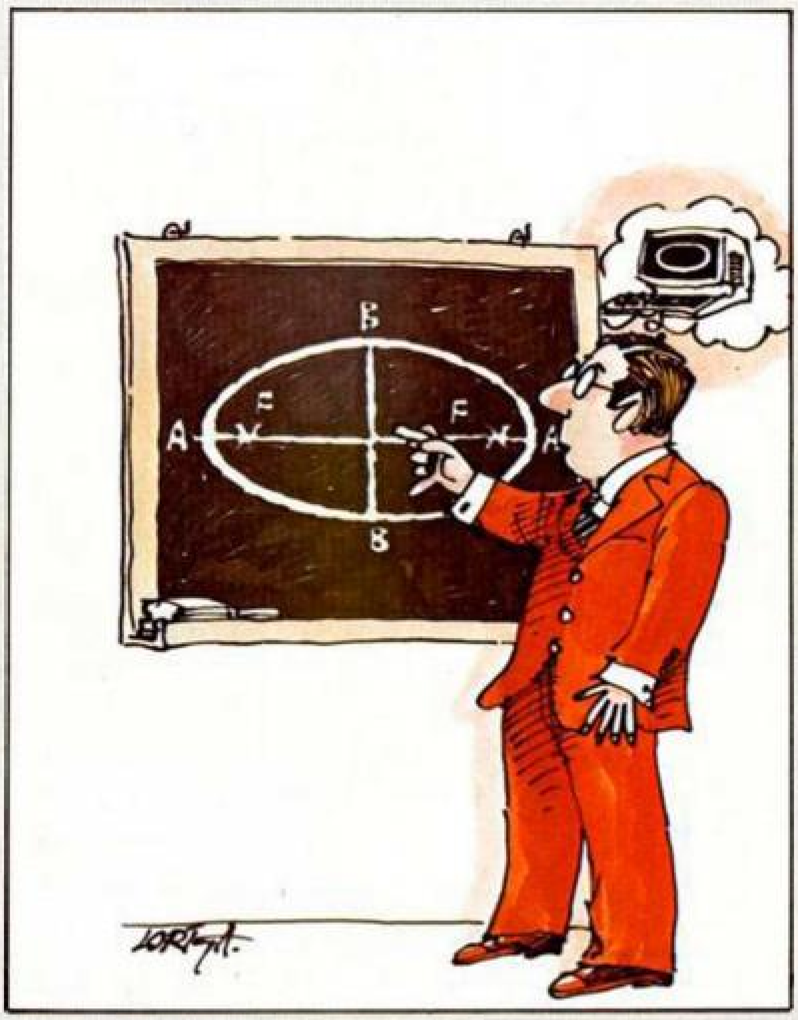
Programa 1. Rutina en código máquina

```
  10 FOR f=0 TO 27: READ a: POKE
32768+f,a: NEXT f
  20 DATA 42,118,92,229,22,0,58,
176,92,95,225,126,35,229,38,0,11
1,254,0,40,5,205,181,3,24,234,19
3,201
```

El valor 0 es el que indica al programa 1 cuando debe devolver el control al Basic, por lo tanto, mientras no se encuentre un cero, seguirá emitiendo sonidos.

El programa 2 es una demostración de cómo usarlo. En este caso DIR vale 990 o, lo que es lo mismo, emplea los valores de la ROM, desde la dirección 990 hasta que encuentre el valor cero.

La duración DUR puede ir desde 0 hasta 255, siendo 0 la duración más corta.



Para incluir esta rutina en nuestro programa basta listar el programa 1 (que además es reubicable) y la línea 9000 del programa 2.

Programa 2

```
   1 LET sound=9000
  10 LET dir=990: LET dur=0: GO
SUB sound
  20 GO TO 10
8999 STOP
9000 RANDOMIZE dir: POKE 23728,d
ur: RANDOMIZE USR 32768: RETURN
```

Si queréis conseguir otras posibilidades probar a cambiar la línea 10 por ésta:

```
  10 LET dir=60000: LET dur=0: G
O SUB sound
```

y poner una de estas dos líneas:

```
   2 LET a=255: FOR f=60000 TO 6
0100: POKE f,a: LET a=a-1: NEXT
f
```

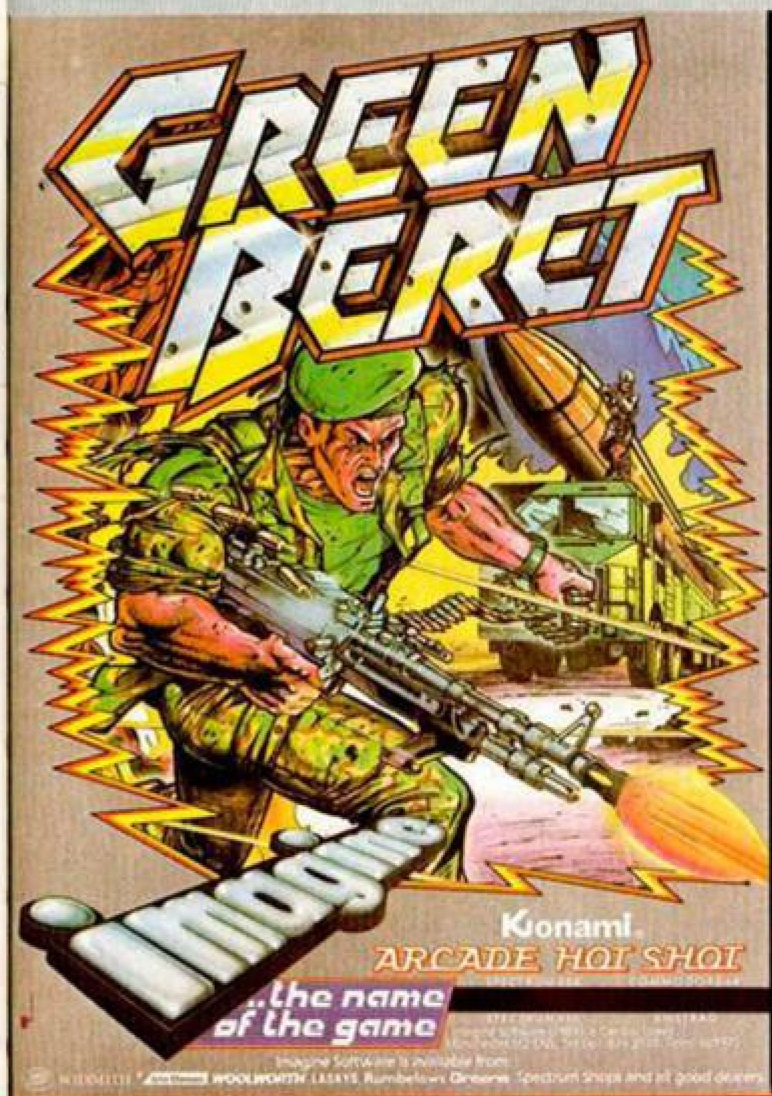
```
   2 LET a=255: FOR f=60000 TO 6
0100 STEP 2: POKE f,a: POKE f+1,
256-a: LET a=a-1: NEXT f
```

ERBEdades

"V"

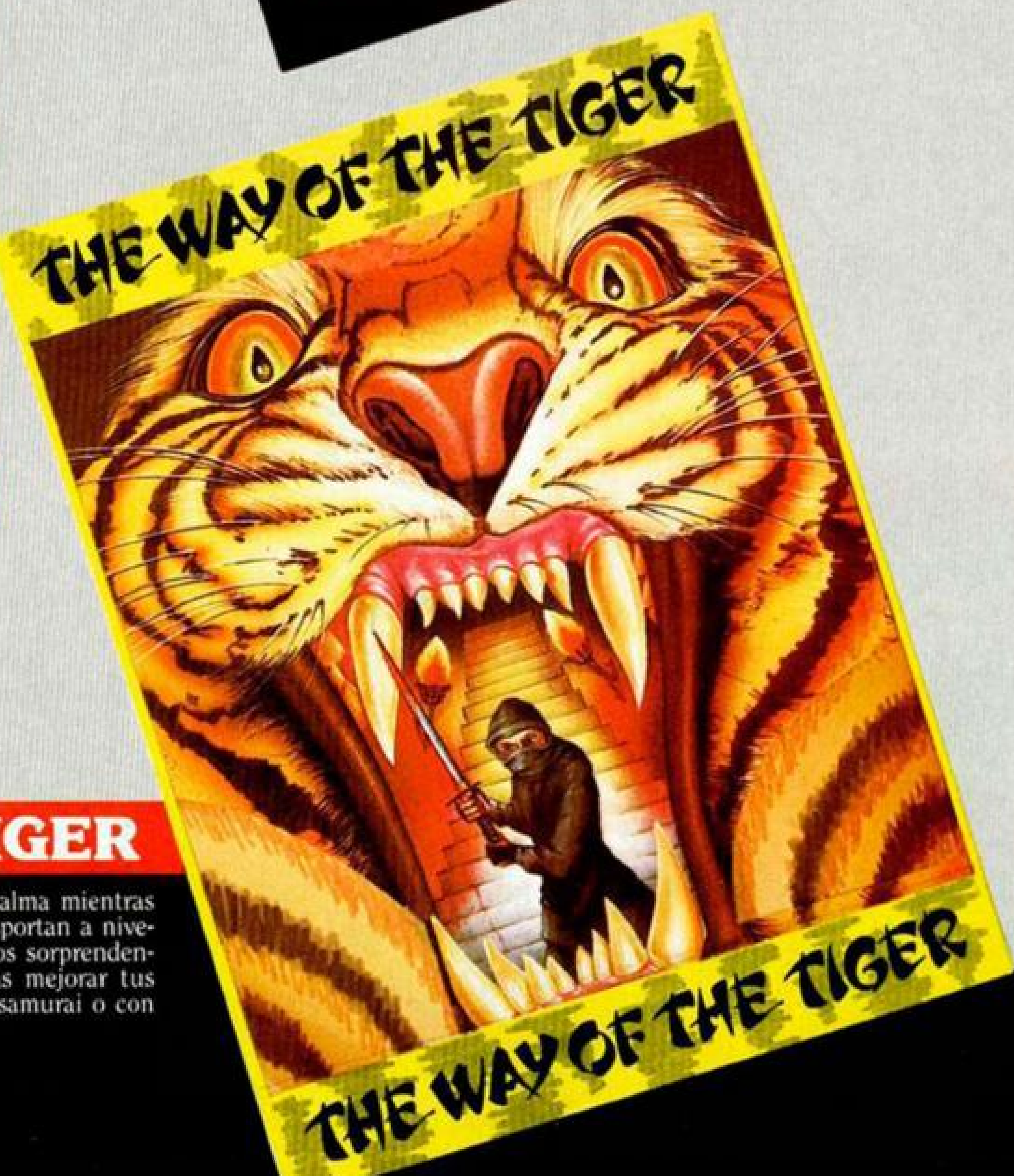
La serie de ciencia-ficción televisiva invade tu ordenador. La Tierra es visitada por alienígenas de aspecto humano. Si embargo, estos reptiles pronto revelarán su objetivo final... conquistar la Tierra y esclavizar a la raza humana.





GREEN BERET

Green Beret (boina verde), es el programa de mayor acción que hayas visto jamás en un ordenador.
Decir Green Beret es decir un hombre perfectamente entrenado para la lucha. Prepárate...



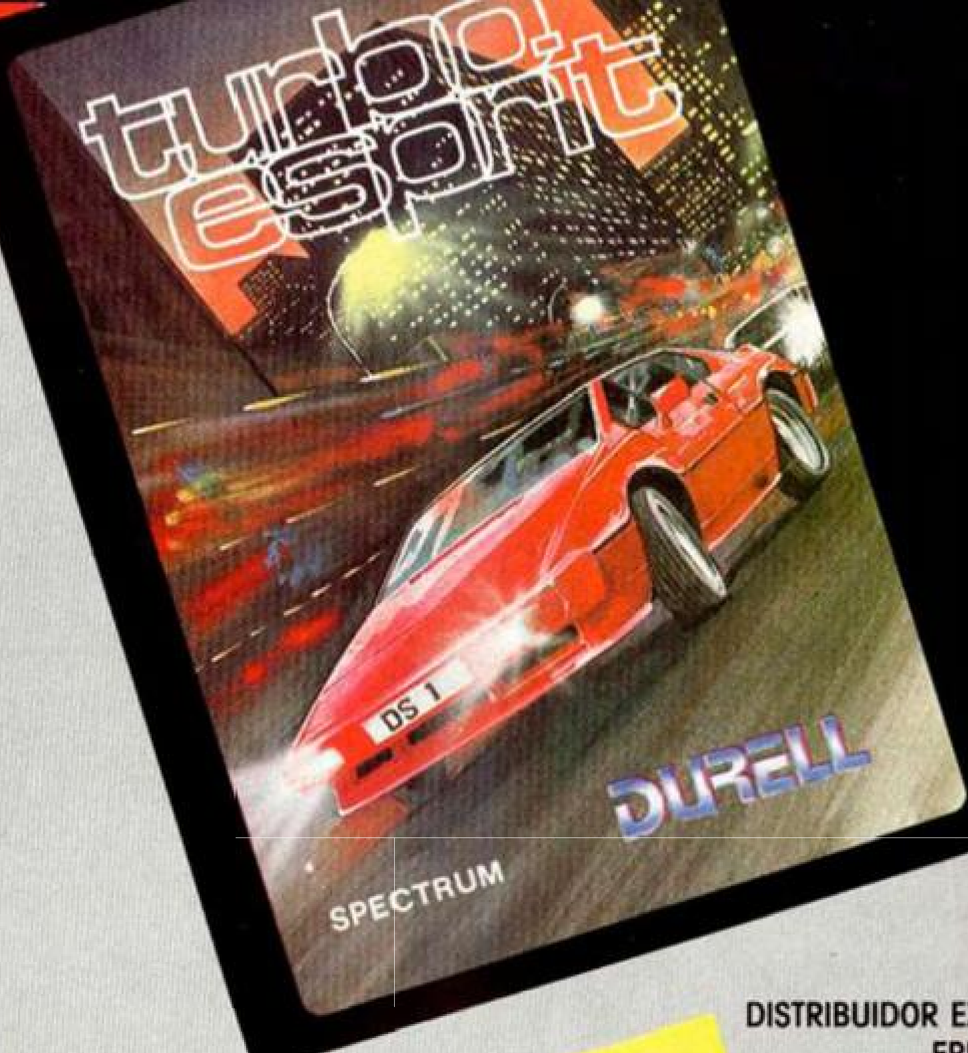
THE WAY OF THE TIGER

Entra en el mudo de los samurais. Mantén la calma mientras el movimiento y las rutinas de combate te transportan a niveles que nunca pensaste posibles. Experimenta los sorprendentes efectos del "Triple Scroll" mientras intentas mejorar tus técnicas de lucha cuerpo a cuerpo, con espada samurai o con mil posibilidades más.

para que juERBEs

TURBO ESPRIT

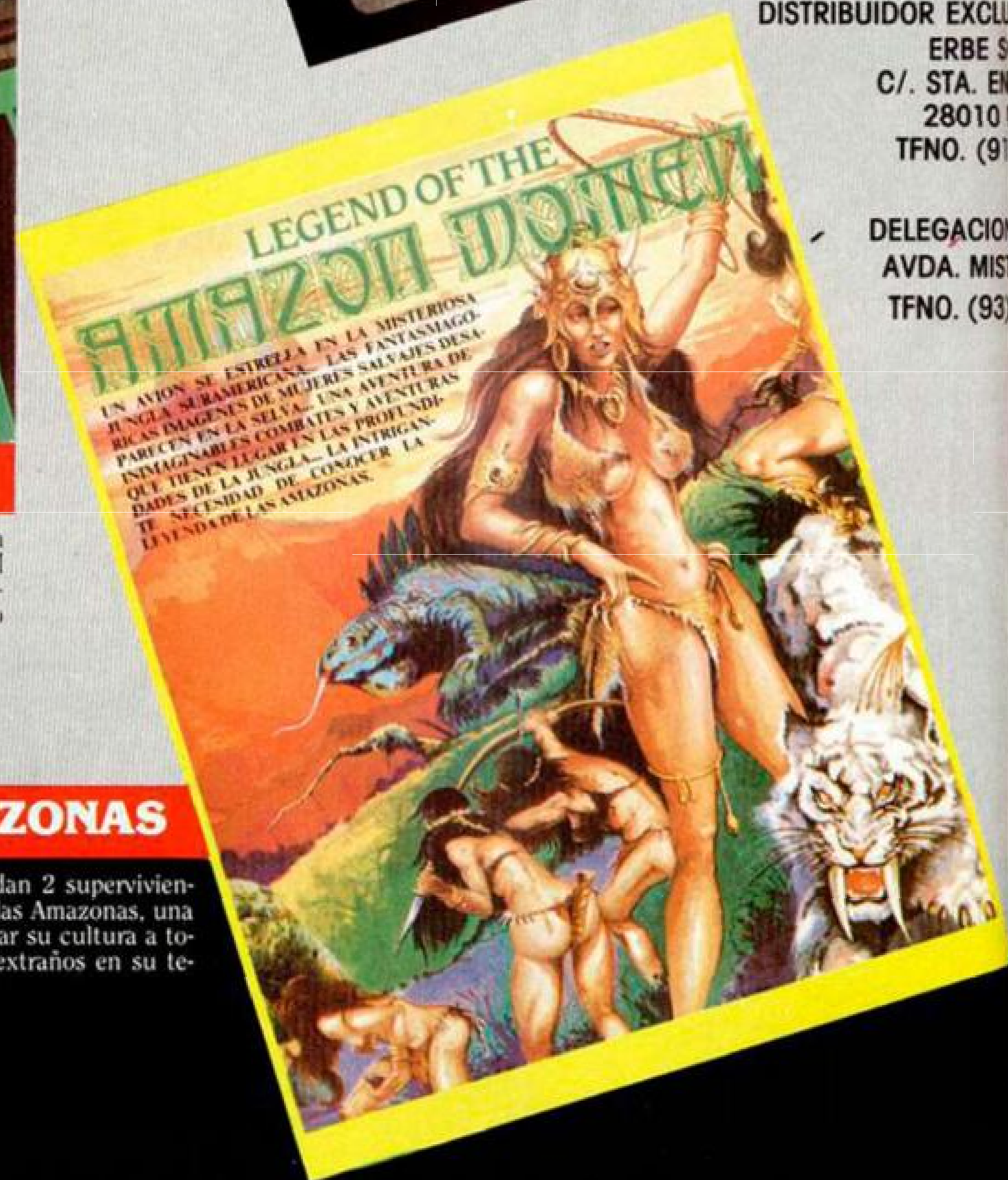
Tu misión: vigilar y cuidar el cumplimiento de la ley que se ve amenazada por una terrible banda de delincuentes que han hecho del tráfico de narcóticos su negocio más rentable.
Tus medios: un Lotus Turbo Sprit dotado de uno de los máximos adelantos técnicos y con el que deberás patrullar por calles y avenidas.





CYBERUN

Prepárate para una carrera sin límites a través del espacio con este nuevo juego de Ultimate. Los impresionantes gráficos y el inimitable estilo al que esta compañía nos tiene acostumbrados te transportarán a una auténtica aventura espacial a bordo del Cyberun.

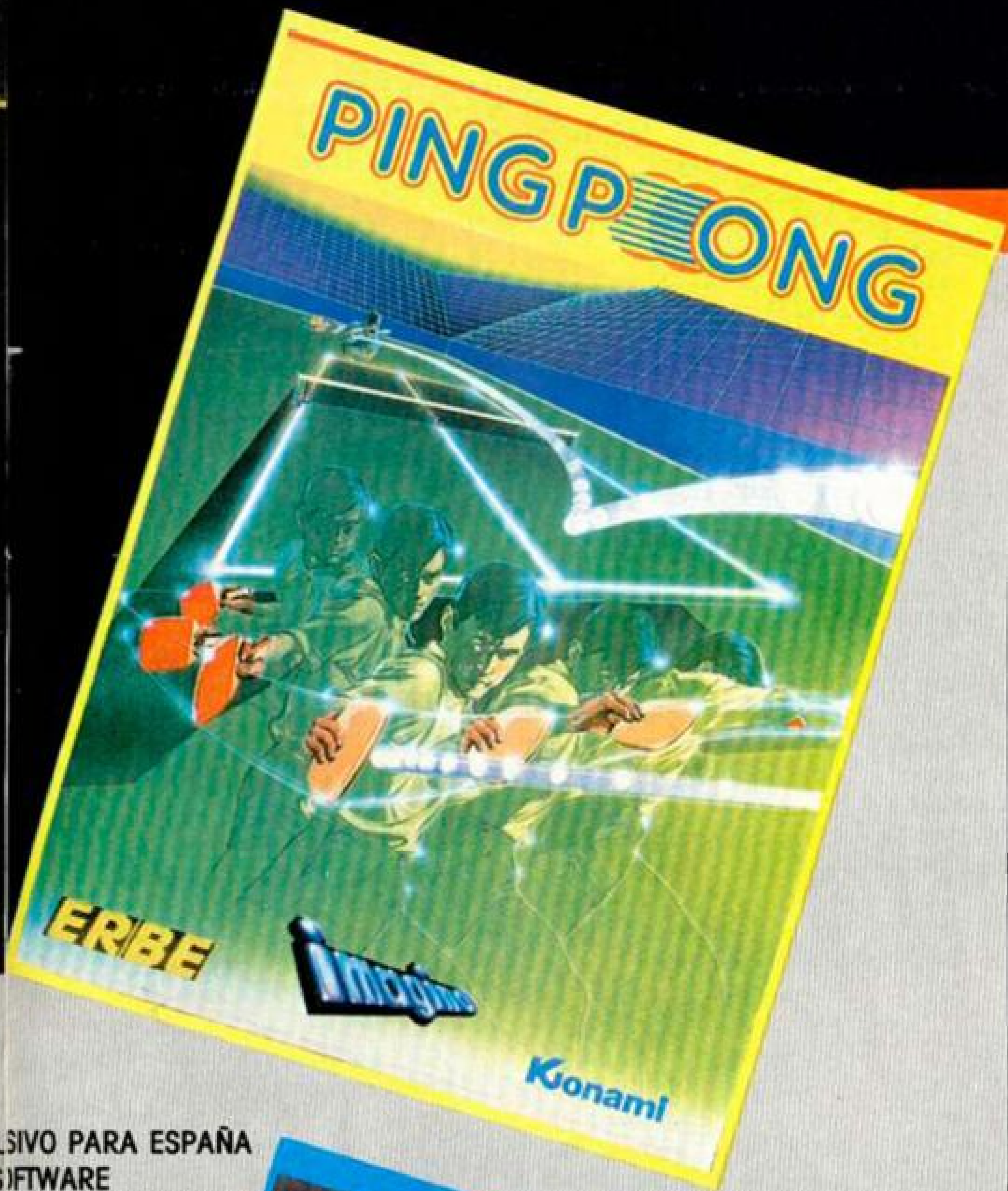


LA LEYENDA DE LAS AMAZONAS

Un avión se estrella en la jungla. Sólo quedan 2 supervivientes: lady Wilde y su hija que es raptada por las Amazonas, una raza de mujeres míticas que quieren conservar su cultura a toda costa y que no admiten la presencia de extraños en su territorio.

DISTRIBUIDOR EXCLUS
ERBE S
C/. STA. EN
28010
TFNO. (91)

DELEGACION
AVDA. MIST
TFNO. (93) 4

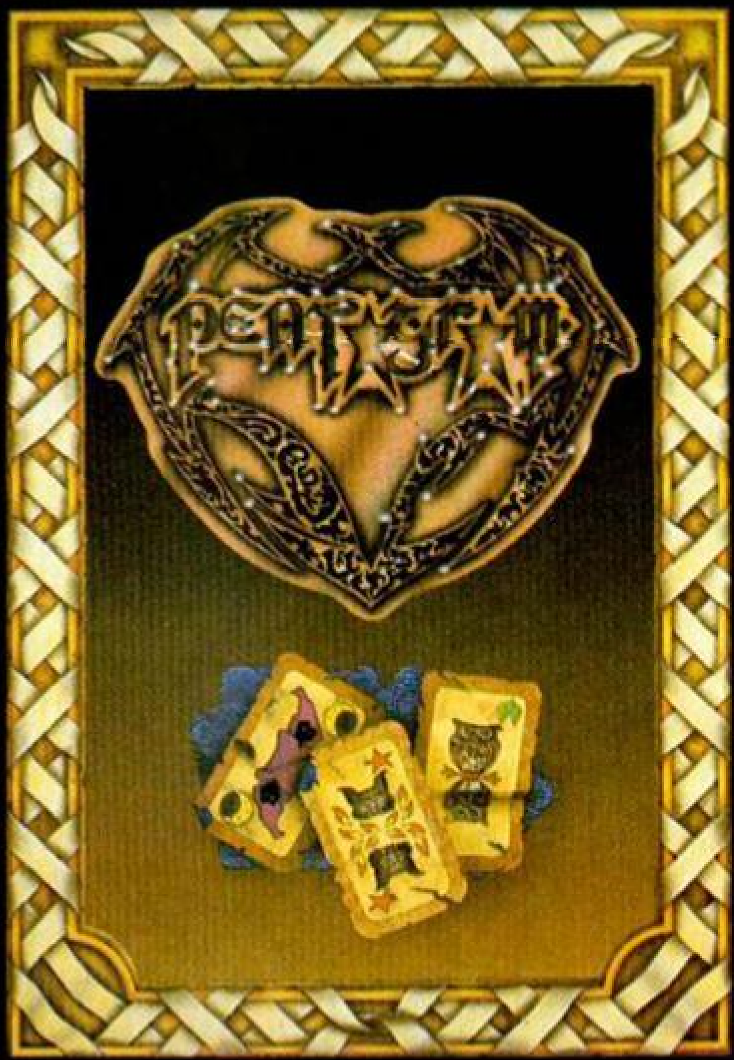


PING-PONG

La gran sorpresa. Gráficos increíbles, movimiento super-rápido, podrás efectuar las mismas jugadas que si tuvieras la paleta en tu mano.
Botes, rebotes, efectos, dejadas, saques, cortadas, mates... todo es posible con esta maravilla llamada "Ping-Pong".

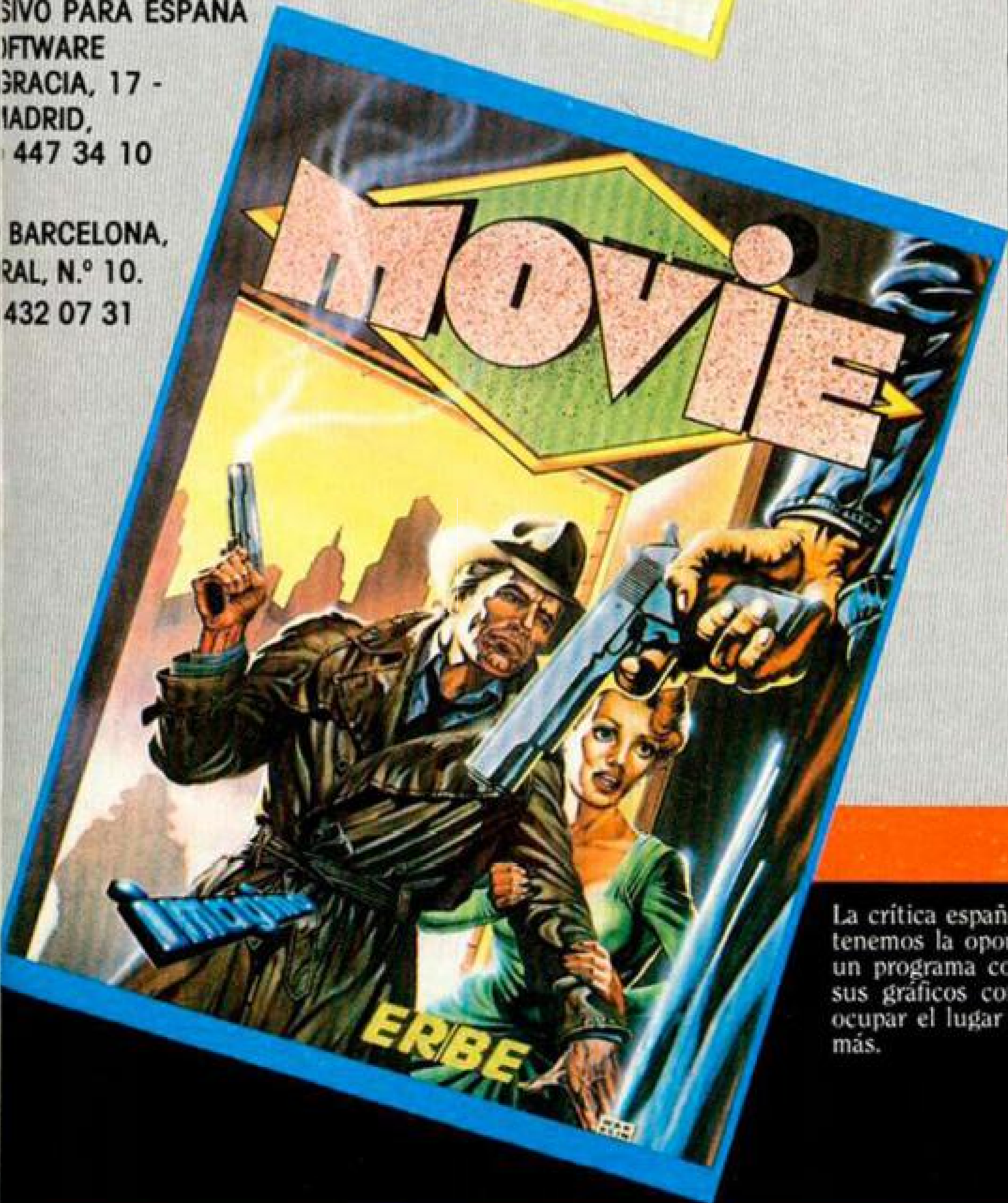
LSIVO PARA ESPAÑA
S)FTWARE
NGRACIA, 17 -
IADRID,
1 447 34 10

N BARCELONA,
SRAL, N.º 10.
)432 07 31



PENTAGRAM

El juego sorpresa de Ultimate que debes descubrir tú mismo.
... La continuación de "Saber Wulf".



MOVIE

La crítica española ha dicho de este juego: "Todos los días no tenemos la oportunidad de disfrutar en nuestro ordenador de un programa como éste. "Movie" es un juego que, tanto por sus gráficos como por su gran originalidad, está llamado a ocupar el lugar de una auténtica estrella." No hace falta decir más.

2112. Videoaventura. Design Design

EL IMPERIO DE LOS ORDENADORES

La tecnología ha avanzado vertiginosamente en las últimas décadas. Ni los más imaginativos escritores de ciencia-ficción hubieran podido pensar que hoy, un día cualquiera del año 2112, todo lo que rodea al hombre, incluso sus animales de compañía, se encuentra bajo el control de los ordenadores.

Efectivamente, el desarrollo tecnológico que ha sufrido nuestro planeta en el transcurso de este siglo, ha desbordado cualquier previsión. La cibernética ha inundado todas las facetas de la vida y podemos afirmar que prácticamente no queda ni una sola actividad humana que no sea realizada, directa o indirectamente, a través de un ordenador. Los ciudadanos hacen sus compras mediante un ordenador, trabajan delante de un ordenador y necesitan de un ordenador para divertirse.

Sin embargo, a pesar de que vivimos en una época en la que poco espacio queda para los resentimientos y las emociones, las máquinas nunca podrían llegar a anular la propia esencia del hombre. Como en cualquier otro período de la historia, cada individuo posee sus propias inclinaciones, sus propias virtudes, sus debilidades.

Por esta razón, incluso en momentos en los que parece que el mundo vive un período de prosperidad, siempre se corre el riesgo de que aparezca algún paranoico que, por sus propios intereses, rompa en un instante este frágil equilibrio.

Esto es lo que ha ocurrido en nuestra ciudad en los últimos días. Un mercenario descontrolado perteneciente al grupo terrorista Hacker Mainhoff, que responde al nombre de Nagaer, ha conseguido adentrarse en las instalaciones de la Sideral Information of Digital Accountings (S.I.D.A.) y tras burlar los diferentes mecanismos de seguridad, ha logrado llegar hasta la sala donde se encuentra el Ordenador Central encargado de controlar el destino de todos los ciudadanos de la Tierra y las órdenes que salen por sus terminales son las que rigen la política y el gobierno de cada uno de los países.

Pues bien, debido a unos cambios que Nagaer ha conseguido realizar en su sistema operativo ha convertido a este ordenador, que había sido un demócrata de toda la vida, en un verdadero fascista cuyo único propósito es el de acabar con el régimen de libertades que hasta ahora reinaba en nuestro planeta.

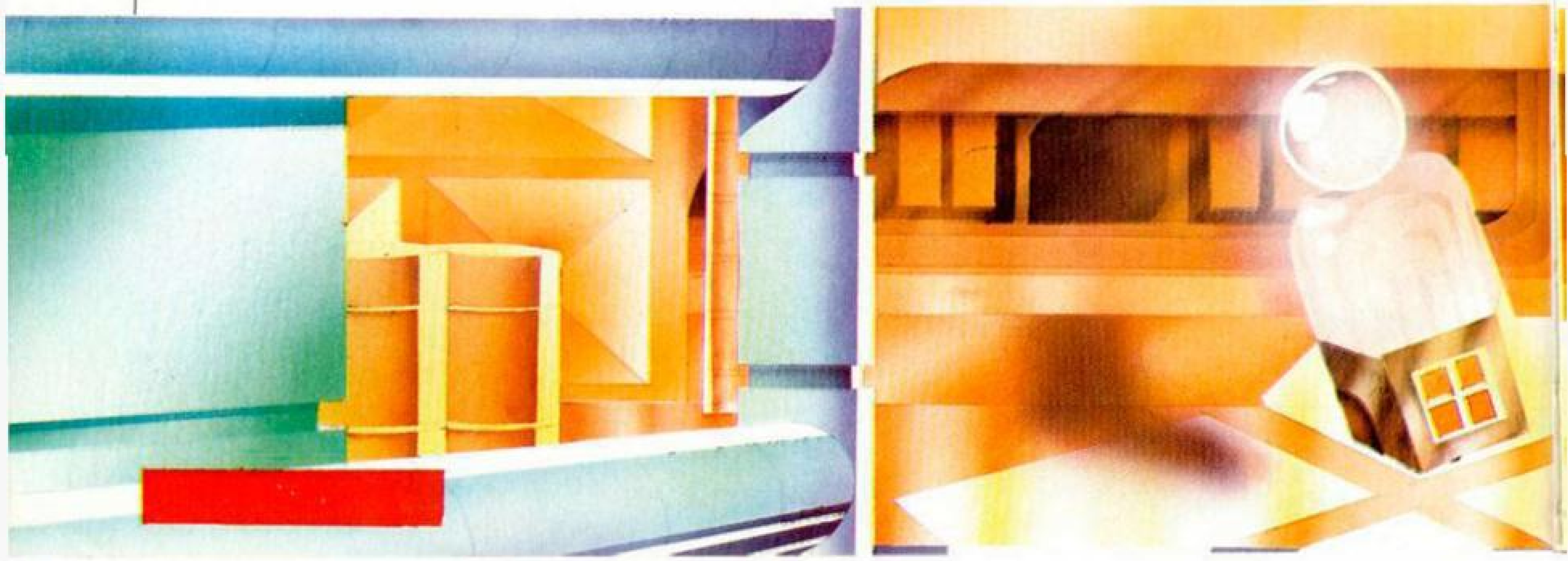
Aquí es donde tú haces aparición, bueno, tú y tu perro mecánico. Vuestra misión será la de encontrar las piezas del Código de Funcionamiento que permiten que, colocadas en el orden exacto, el Ordenador Central vuelva a su estado normal y recobre definitivamente su cordura. Estas piezas se encuentran ocultas entre los innumerables edificios, estancias y pasadizos de la ciudad, la cual está dividida en varias zonas que se comunican entre sí.

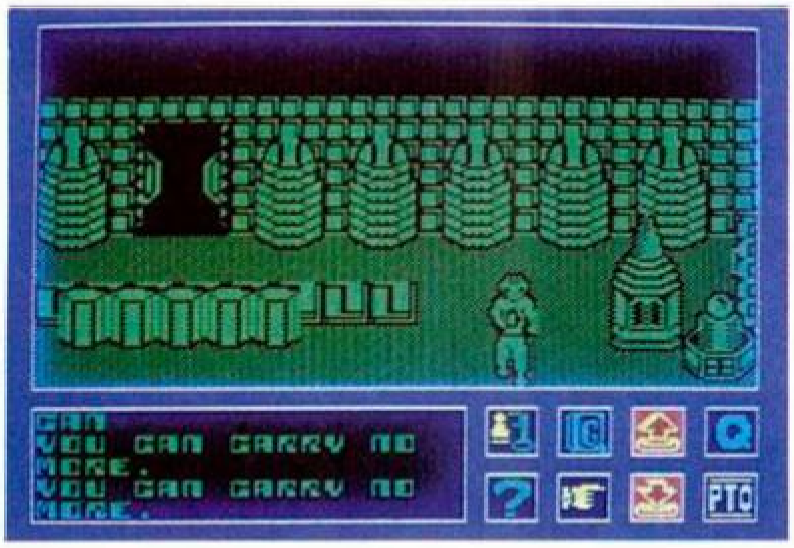
Pero, a pesar de que la misión es bastante complicada, por lo menos cuentas con la inestimable colaboración de Poody, un robot-perro que te acompañará donde quiera que vayas —aunque algunas veces tendrás que esperar por él—, y que te servirá de gran utilidad no sólo para transportar los objetos que vayáis recogiendo sino también para sacarte de apuros en numerosas ocasiones en las que peligra tu integridad física. En una palabra, que a partir de este momento tú y Poody vais a ser como Tintín y Milú, pero a lo moderno.

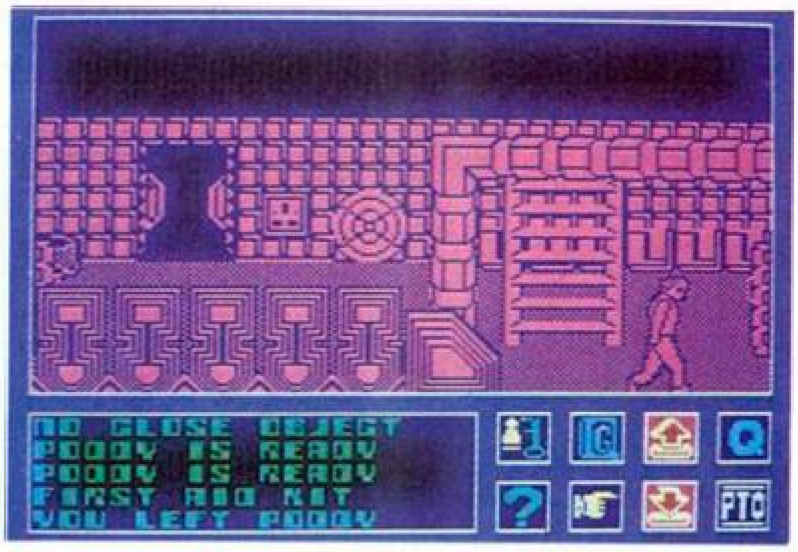
Juntos recorreréis los metalizados rincones de la city y juntos tendréis que encontrar los objetos imprescindibles para finalizar con éxito vuestro libertario cometido: latas de comida que os ofrezcan las energías necesarias, linternas que os iluminen el camino, llaves que os abran puertas misteriosas, y lo que es más importante, las piezas del código de funcionamiento.

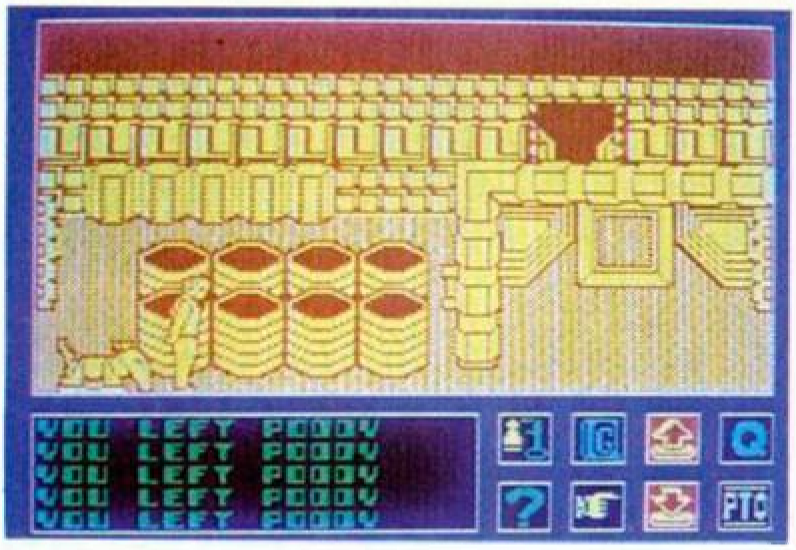
Y esto es cuanto os podemos contar en lo referente a la trama de este 2112 AD que como comprobaréis, es bastante interesante; pero acerca de cómo se juega, todavía quedan algunas cosas que comentar.

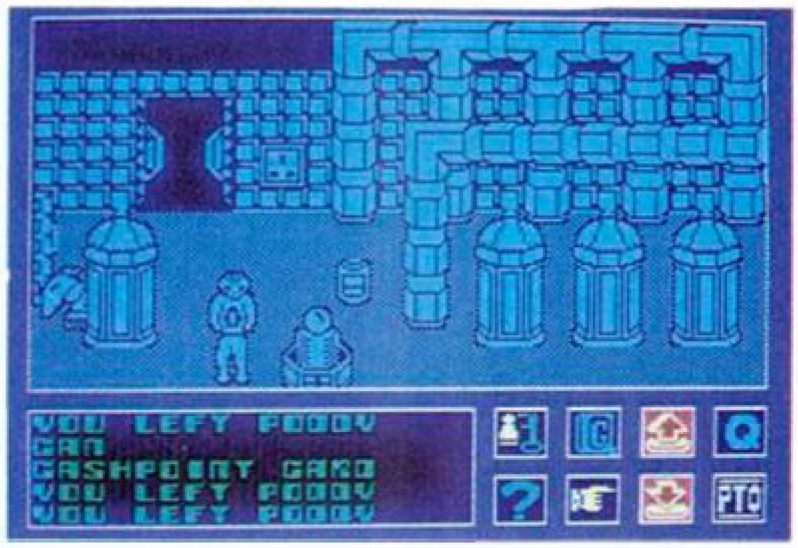
Por ejemplo, obligado es señalar que el programa sigue la línea puesta muy de moda últimamente de llevar el control de los personajes y de sus acciones a través











de un sistema de iconos, (tal y como se lleva a cabo en juegos como Movie o la mayoría de los diseñadores de gráficos). Sin embargo, 2112 le ha intentado sacar el máximo rendimiento a esta circunstancia y a través de sus ventanas se puede acceder, nada más y nada menos, que a 27 acciones diferentes. Estas son de la más variada naturaleza y abarcan desde controles para seleccionar joysticks, grabar, cargar, y otros detalles relativos a cuestiones puramente técnicas, a otras puramente relacionadas con el desarrollo del juego, como pueden ser: coger, soltar, inventarios, estado de salud del perro, del personaje, entregar cosas a Poody, coger objetos que él lleve, primeros auxilios, comer, abrir, y así hasta un total de posibilidades más que considerables.

Todas estas circunstancias hacen que de por sí el programa presente un alto grado de interés, pues, por supuesto, podemos deducir que no va a resultar monótono en absoluto.

De esta manera se ha conseguido un programa adictivo y variado, que además cuenta con la ventaja de que los gráficos, el colorido y el scroll de la pantalla son realmente originales y atractivos.

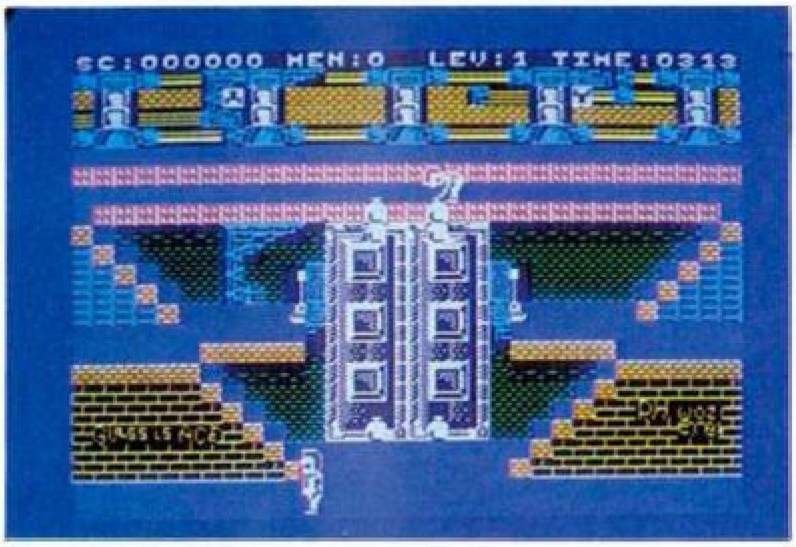




SCHIZOFRENIA • Arcade • Quicksilva

LA PERSECUCION DE ALPHONSE II

Schizofrenia es un arcade con una estructura general bastante simple, pero en realidad, como el propio nombre indica, se trata de un juego para volverse loco. Tan sólo posee cinco pantallas diferentes en las cuales se desarrolla la totalidad de la acción, sin embargo, cada una de ellas presenta una dificultad, a nuestro juicio, excesiva.



La primera de las pruebas consiste en algo aparentemente trivial, pero que, a la postre, nos hará desesperar. Consiste en que Alphonse, el protagonista, debe intentar abrir una enorme puerta, la cual está asegurada con cuatro enormes cerrojos. Hasta aquí todo perfecto, pero el problema empieza cuando Alphonse II, androide gemelo a nuestro héroe, nos comience a perseguir por todas partes y vaya deshaciendo todo lo que hemos conseguido con un esfuerzo supremo.

Así, de esta forma, podremos tirarnos horas y horas dando vueltas por la pantalla, subiendo y bajando las escaleras y tratando de quitarnos de encima a este implacable perseguidor. Pero todo será en vano, pues seguramente, después de todo, no conseguiremos sacar nada en claro.



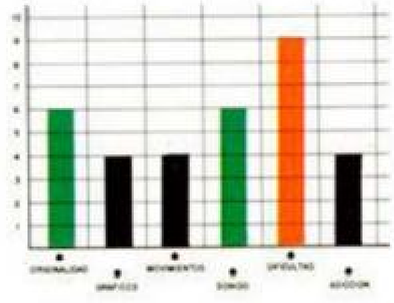
Sin embargo, existen, como antes dijimos, otras cuatro misiones más. La segunda de ellas consiste en limpiar seis cajas amarillas que aparecen moviéndose en una plataforma. Para llegar hasta ellas tendremos que hacer uso de unos ascensores colocados a tal efecto. Pero no os hagáis ilusiones, seguro que no conseguiréis limpiar ni una.

Ahora Alphonse, poniéndonos en el caso utópico de que ha conseguido realizar la fase segunda, deberá continuar con sus afanosos servicios de limpieza, para pasar posteriormente a áreas superiores en las cuales deberá volver a abrir puertas y desactivar enchufes y conectores.

El caso es que a pesar de que varíen los escenarios, el rollo del juego consiste en volverse mico intentando descubrir el camino exacto y los movimientos adecuados que nos permitan acabar nuestro cometido y quitarnos de encima al pesado de Alphonse II, lo cual, insistimos una vez más, resultará poco menos que imposible.

Un detalle curioso de Schizofrenia, por no decir el único, es que junto a las instrucciones aparece un papel que contiene la solución exacta a cada una de las fases, por lo que cuando nos encontremos demasiado desesperados, no tendremos más que raspar la pintura que cubre estos intrincados secretos.

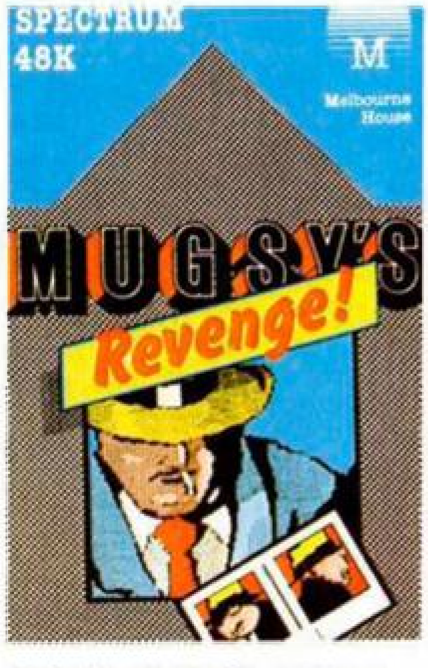
¡Ah!, gráficamente y en cuanto a los movimientos: pobre, pobre.



MUGSY'S REVENGE • Estrategia • Melbourne House

LA GUERRA DEL WHISKY

Nos encontramos en el año 1919. Hace ya algunos días que Mugsy, uno de los principales reyes del mundo del hampa, salió de la cárcel donde estuvo pasando una larga temporada a la sombra.



Las cosas han cambiado mucho desde sus primeras incursiones en el turbio mundo del crimen. Las calles y ciudades de América no son como antes. En los rostros de la gente se puede apreciar la incertidumbre, el temor... El Gobierno ha implantado la total prohibición del consumo y tráfico de alcohol en todo el país y, como siempre, unos cuantos han decidido sacar el máximo partido de esta conflictiva situación. Los mafiosos han creado un turbulento mercado negro y cada vez se están haciendo más y más poderosos, más y más temidos.

Mugsy no ha hecho más que comenzar su incipiente



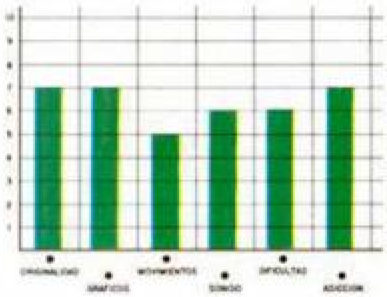
carrera, pero con su astucia seguro que consigue controlar la mayor parte del mercado del whisky. De todas formas deberá permanecer siempre alerta, pues ésta es una carrera a muerte, en la que no se puede parar a mitad del camino.

Así pues, si decides acompañar a Mugsy en esta peligrosa aventura, nada te pillará ya por sorpresa. Sabes que esto no va a consistir tan sólo en comprar y vender. No, ni mucho menos. Además de preocuparte de los aspectos puramente de negocios, deberás estar al tanto de muchos otros detalles: tendrás que buscarte unos buenos «amigos» que te protejan en todo momento, en algunas ocasiones tendrás que hacer algún que otro soborno a la policía, mantener contentos a tus muchachos, estar al día con tus contactos, deshacerte de la mercancía en los momentos que lo requieran, vigilar a tus confidentes, eliminar a quien te moleste... Pero lo más importante es que todo lo tendrás que meditar muy detenidamente y siempre deberás actuar con la máxima precaución, pues el menor fallo, el menor descuido, puede resultar fatal para vuestra salud.

Mugsy es, por tanto, un juego de estrategia en cuyo desarrollo deberemos ir afrontando diferentes tipos de situaciones que van desde partes de diálogo con el ordenador, hasta escenas animadas, pasando por alguna que otra fase de arcade en la cual tendremos que demostrar nuestras habilidades como pistoleros.

Por todo esto, y porque los gráficos, aunque excesivamente escasos, son de lo más original y artístico de cuanto hemos visto, Mugsy es un excelente programa para los que disfrutan pensando y maquinando frente al ordenador.

Un fallo, el de siempre: está en inglés y nos atreveríamos a asegurar que no va a ser traducido al castellano. Por desgracia, ya estamos acostumbrados a todo.



TURBO ESPRIT • Simulador • Durell

UNA VERTIGINOSA CARRERA

Durell nos presenta, tras su último y espectacular éxito —Saboteur—, su programa de más reciente realización: Turbo Esprit.



Este nuevo juego, si bien posee un nivel de calidad muy similar al de Saboteur en cuanto a gráficos, movimientos y programación en sí se refiere, la verdad es que resulta bastante menos adictivo que el primero.

Turbo Esprit es un programa de simulación en el cual nos sentamos al mando de un Lotus Turbo y nos introducimos de lleno en el complejo y enrevesado laberinto de la ciudad de Manhattan.



Pero éste no va a ser un viaje de placer, ni tampoco una frenética carrera en la que tendremos que luchar por conseguir llegar los primeros a la meta. En esta ocasión se trata de un asunto de drogas.

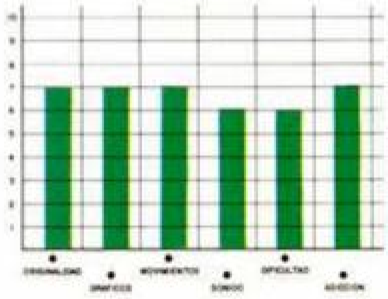
A bordo de nuestro Lotus representaremos el papel de un agente encargado de desarticular una banda de traficantes que últimamente viene actuando por la ciudad y que está causando verdaderos estragos.

La misión específica consiste en capturar a los cuatro coches que realizan el reparto de las mercancías y otro vehículo blindado que es el encargado de suministrarles las drogas. Para la localización de estos coches disponemos además de un plano general de la ciudad, de una pequeña terminal que nos transferirá la información de nuestra situación y de la de nuestros perseguidos. Una vez que consigamos encontrarnos con alguno de los coches, estaremos en disposición de, bien acabar directamente con él o bien de apresar a los ocupantes, lo que se consigue disparando contra ellos o chocándonos por detrás, respectivamente.



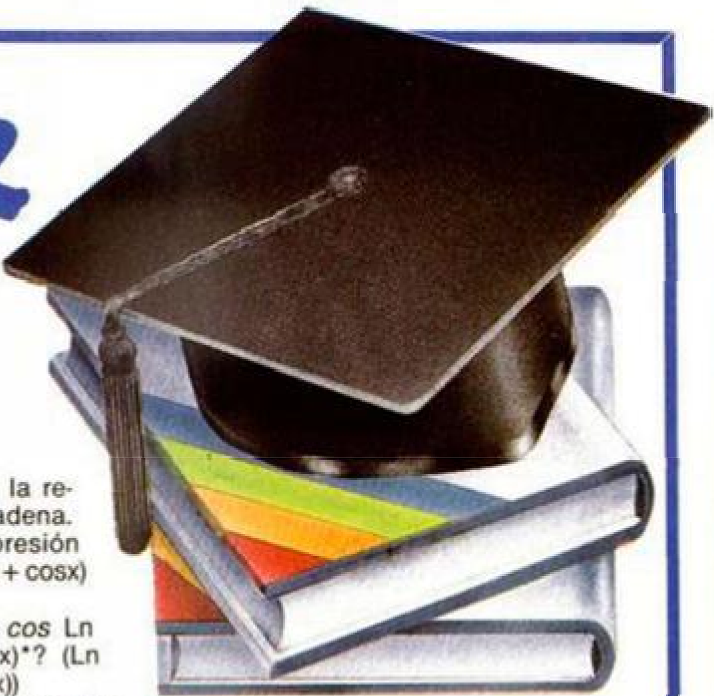
El juego en sí no está mal del todo, pues tiene su parte de acción y su parte de estrategia, pero tampoco resulta excesivamente divertido.

Los aspectos gráficos están muy bien confeccionados ya que, aunque de manera muy lineal, consiguen imprimir bastante bien un efecto de movimiento rápido. Este es, sin duda, su mayor atractivo.



Profesor particular

Arturo LOBO y J. J. LEON



DERIVADAS 1

Esta semana publicamos la primera parte del programa «Derivadas» que por su extensión no os podemos ofrecer en un solo número.

«Derivadas» es capaz de obtener la derivada de cualquier función escrita utilizando las operaciones matemáticas del Spectrum.

Principalmente está enfocado a la enseñanza y no para solucionar los deberes de casa. Tiene varios niveles según el grado de conocimiento de cada usuario. En el nivel más básico calcula la derivada de una función paso a paso, muestra en la pantalla la operación o función que va a derivar y os pide permiso para continuar.

Además de obtener derivadas puede plantearos un problema, de dificultad a elegir, o comprobar la solución que hayáis obtenido vosotros, o hallar la derivada en un punto.

Otra aplicación muy importante que puede realizar es obtener la ecuación de la recta tangente en un punto a la función que le deis, tanto si está en explícitas (y = f(x)) o en implícitas (f(x,y) = 0).

Cuando el programa os pida la función para obtener la recta, tangente, derivar, etc., basta con dar ENTER si la función ya está dentro (por ejemplo si es la que os ha planteado como problema).

La notación que utiliza así como el método de derivar es algo particular, lo cual será una ventaja si tenéis problemas con otros métodos.

Para denotar la derivada pone una interrogación delante de lo que falta por derivar. Así, lo que hay escrito en un papel es d/dx (función), es en la pantalla? (función).

El método de derivación está pensado para ser didáctico, deriva de varias pasadas por la expresión y muy ordenadamente, derivando sólo una cosa cada vez, lo cual es muy eficaz para aprender la regla de la cadena. Así, la expresión sen Ln (x↑2 + cosx) haría:

1.ª pasada: *cos* Ln (x↑2 + cos x)*? (Ln (x↑2 + cos x))

2.ª pasada: cos Ln (x↑2 + cosx)*1/(x↑2 + cos x)*?(x↑2 + cosx)

3.ª pasada: cos Ln (x↑2 + cos x)*1/(x↑2 + cos x)*(?(x↑2) + + ?(cos x))

4.ª pasada: cos Ln (x↑2 + cos x)*1/(x↑2 + cos x)*(2*x↑(2-1) + + (—sen x))

Tenemos que advertir que el programa no corre si está en mayúsculas. También que debido a que el Spectrum no realiza operaciones como (—2)↑2 cuando aparezca la función «elevar» el programa puede dar error si queremos calcular la derivada en un punto o comprobar vuestra solución.

Para derivar una función de x elevada a otra utiliza la fórmula que se obtiene derivando por el método logarítmico.

Los productos los deriva todos a la vez y las divisiones también, por ejemplo: ?(f(x)*g(x)*h(x)) = ?(f)x))*g(x)*h (x) + f(x)*?(g(x))*h(x) + f(x)*g(x) *?(h(x)) y esta fórmula es generalizable a cualquier número de factores. En cuanto a las divisiones, las opera como si fuera el producto por el inverso, o sea, ?(f(x)/g(x)) = ?(f(x)*(1/g(x)).

Si tenéis alguna duda podéis consultar con vuestro profesor, no obstante, la próxima semana os ofreceremos junto con la segunda parte del programa una guía de cómo aprender a derivar.

```
1997 DEF FN f(x)=VAL f$
1998 DEF FN d(x)=VAL e$
1999 DEF FN r(x)=asd+INT (RND*x)
2000 BORDER 5: PAPER 5: INK 0: CLS : LET asd=1: LET i$="x"
2005 LET ense=0
2010 GO TO 3000
2020 GO SUB 9600: LET g$=e$
2023 PRINT AT 0,0;"Tienes que derivar la siguiente funcion "
2025 PRINT AT 4,0;e$: GO TO 9750
2030 INPUT "Que funcion quieres derivar""f(x)= "; LINE u$
2032 IF u$<>"" THEN LET g$=u$
2035 LET e$=g$: GO TO 8000
2040 PAUSE 10: PRINT AT 15,0;"Niveles""0- Sin ayuda""1- Alto""2- Medio""3- Basico"
2045 LET j$=INKEY$: IF j$="" THEN GO TO 2045
2046 LET ense=(j$="1")+2*(j$="2")+3*(j$="3")
2048 PRINT AT 21,0;"OK": PAUSE 10: RETURN
2050 INPUT "En que punto quieres la derivada""  x0= ";x0
2055 PRINT AT 15,0;"Su valor es ";FN d(x0): GO TO 9750
2060 INPUT "Dime tu derivada"' LINE f$
2065 LET ert=1.234567: FOR t=1 TO 5: GO TO 2067+(ABS (FN f(ert*t)-FN d(ert*t))<1e-6)
2067 BEEP 1,0: PRINT AT 15,0;"Lo siento, es incorrecta": GO TO 9750
2068 NEXT t: PRINT AT 15,0;"En efecto,esta bien": BEEP .1,5: BEEP .2,20: GO TO 9750
2070 PRINT AT 17,0;"La funcion esta en explicitas(e)""y=f(x)    , o bien en implicitas(i) f(x,y)=0 ?"
2075 DIM x(2)
2080 INPUT "(e/i)  "; LINE j$: LET jota=10*(j$="i")
2083 DATA "f(x)= ","y=f(x)"
2084 DATA "f(x,y)= ","f(x,y)=0"
2085 CLS : RESTORE 2083+(j$="i"): READ u$: INPUT (u$); LINE u$
2090 IF u$<>"" THEN LET g$=u$
2093 LET e$=g$
2095 INPUT "Dime el punto (a,b) de la curva""x= ";ax1;"  y= "; LINE o$
2100 IF o$<>"" THEN LET by1=VAL o$: GO TO 2105
2103 LET x=ax1: LET by1=VAL g$
2105 PRINT AT 2,0;"La ecuacion general de la recta""tangente en el punto (a,b) a la";
2110 PRINT " funcion ";: READ u$: PRINT u$;" es ": GO TO 2120+jota
2120 PRINT AT 9,8;"y-b = (f'x)*(x-a)": GO TO 2140
2130 PRINT AT 8,11;"y-b";AT 8,20;"x-a"
2132 PRINT AT 10,11;"f'x";AT 10,20;"f'y"
2135 PRINT AT 9,16;"= -": PLOT 87,100: DRAW 25,0: PLOT 160,100: DRAW 24,0
2140 LET imp=0: PRINT AT 16,0;"Primero derivo la funcion y    despues hallo su valor en    el punto": GO SUB 9750: GO TO 2160+4*jota
2150 BEEP .1,40: PRINT AT 20,0;"Derivada respecto a ";CHR$ (120+imp): FOR p=1 TO 200: NEXT p
2155 LET e$=g$: LET i$=CHR$ (120+imp): GO TO 8000
2160 GO SUB 2150: CLS : LET x=ax1: LET fx=VAL e$: GO SUB 2500
2165 BEEP .3,5: BEEP .2,30: PRINT AT 10,0;"La recta es ,operando ,""y= ";fx;" * x ";CHR$ (43+2*(by1-fx*ax1<0));" ";ABS (by1-fx*ax1): GO TO 9750
2200 FOR f=0 TO 1: LET imp=f: GO SUB 2150: CLS : LET x=ax1: LET y=by1: LET x(f+1)=VAL e$: GO SUB 2500: NEXT f
2205 LET i$="x"
2210 CLS : BEEP .3,5: BEEP .2,30: PRINT AT 10,0;"La recta es ,operando,""y= ";-x(1)/x(2);" * x ";: LET iuy=by1+x(1)/x(2)*ax1: PRINT CHR$ (43+2*(iuy<0));" ";ABS iuy: GO TO 9750
2500 PRINT AT 5,0;"Esta derivada vale, en el punto (a,b),  f'";CHR$ (120+imp);" = ";VAL e$: GO TO 9750
2990 DATA "MENU GENERAL","1- PLANTEAR UN PROBLEMA"
2992 DATA "2- OBTENER LA DERIVADA","3- APREDIZAJE ?"
2994 DATA "4- DERIVADA EN UN PUNTO","5- COMPROBAR TU SOLUCION"
2996 DATA "6- RECTA TANGENTE EN UN PUNTO"
3000 CLS : RESTORE 2990: FOR t=1 TO 7: READ z$: PRINT AT 2*t+2,2;z$: BEEP .2,3*t
3005 LET i$="x"
3010 NEXT t
3020 LET azx=CODE INKEY$: IF azx>54 OR azx<49 THEN GO TO 3020
3030 CLS : GO SUB 2010+10*(azx-48): GO TO 3000
7500 BEEP .1,3: BEEP .15,0: BEEP .1,20: PRINT AT 0,0;"Funcion ";g$;AT 10,0;"Su derivada es ";e$: GO TO 9750
8000 DEF FN l(x)=x*((x>64)*(x<91)+(x>96)*(x<123)+(x>47)*(x<58))
8001 DEF FN p(x)=3*((x=43)+(x=45))+2*((x=42)+(x=47))+(x=94)
8003 CLS : BEEP .1,5: PRINT AT 0,0;"Funcion ";e$
8005 LET d$=e$: GO SUB 8480: IF pvar=0 THEN LET e$="0": RETURN
8007 LET a$=e$: LET paren=0: GO SUB 8040: LET e$=a$
8008 GO SUB 9690
8010 LET b$="": FOR o=1 TO LEN e$
8012 IF CODE e$(o)<>0 THEN LET b$=b$+e$(o): NEXT o: GO TO 8024
8014 LET np=0: FOR i=o+1 TO LEN e$: IF CODE e$(i)=40 THEN LET np=np+1
8016 IF CODE e$(i)=41 THEN LET np=np-1: IF np=0 THEN GO TO 8020
8018 NEXT i: PRINT FLASH 1;"numero impar de parentesis": STOP
8020 LET nuevao=i+1: GO SUB 9700: LET a$=e$(o+2 TO i-1): LET paren=0: GO SUB 8040
8021 GO SUB 9690: IF a$="1" AND b$<>"" THEN IF b$(LEN b$)="*" THEN LET b$=b$(1 TO LEN b$-1): GO TO 8023
8022 LET b$=b$+a$
8023 LET o=nuevao: IF o<=LEN e$ THEN GO TO 8012
8024 IF b$=e$ THEN CLS : GO SUB 7500: RETURN
8025 IF ense=1 THEN GO SUB 9780
8026 LET e$=b$: GO TO 8010
8039 LET paren=1
8040 DIM a(25,2): DIM b(25): LET ax=1: LET np=0: LET a(1,2)=0: FOR i=1 TO LEN A$: IF CODE A$(i)=40 THEN LET np=np+1: IF np=1 THEN LET ax=ax+1: LET a(ax,1)=i: NEXT i
8045 IF CODE A$(i)=41 THEN LET np=np-1: IF np=0 THEN LET a(ax,2)=i: NEXT i
8047 NEXT i: IF np<>0 THEN PRINT "numero impar de parentesis !!!": STOP
8050 LET a(ax+1,1)=LEN A$+1: IF a(2,1)=1 THEN IF a(2,2)=LEN A$ THEN LET A$=A$(2 TO (LEN A$-1)): GO TO 8039
8055 LET no=0: LET niv=0: LET b(1)=0: FOR j=1 TO ax: FOR i=a(j,2)+1 TO a(j+1,1)-1: LET orden=FN p(CODE a$(i)): IF orden>niv THEN LET niv=orden: LET no=2: LET b(2)=i: GO TO 8062
8060 IF orden=niv THEN IF niv>0 THEN LET no=no+1: LET b(no)=i
8062 NEXT i: NEXT j: LET b(no+1)=LEN A$+1
8065 IF ense>2 THEN GO SUB 9570-10*(niv=0)
8070 GO SUB (8100+100*niv): RETURN
8100 IF a$=i$ THEN LET a$="1": RETURN
8101 IF A$="ABS "+i$ THEN LET a$="SGN "+i$: RETURN
```

INSTRUCCIONES DE PRUEBA DE BITS (I)

Código Fuente	Hexadecimal	Decimal
BIT 0,A	CB,47	203,71
BIT 0,B	CB,40	203,64
BIT 0,C	CB,41	203,65
BIT 0,D	CB,42	203,66
BIT 0,E	CB,43	203,67
BIT 0,H	CB,44	203,68
BIT 0,L	CB,45	203,69
BIT 0,(HL)	CB,46	203,70
BIT 0,(IX+d)	DD,CB,d,46	221,203,d,70
BIT 0,(IY+d)	FD,CB,d,46	253,203,d,70
BIT 1,A	CB,4F	203,79
BIT 1,B	CB,48	203,72
BIT 1,C	CB,49	203,73
BIT 1,D	CB,4A	203,74
BIT 1,E	CB,4B	203,75
BIT 1,H	CB,4C	203,76
BIT 1,L	CB,4D	203,77
BIT 1,(HL)	CB,4E	203,78
BIT 1,(IX+d)	DD,CB,d,4E	221,203,d,78
BIT 1,(IY+d)	FD,CB,d,4E	253,203,d,78
BIT 2,A	CB,57	203,87
BIT 2,B	CB,50	203,80
BIT 2,C	CB,51	203,81
BIT 2,D	CB,52	203,82
BIT 2,E	CB,53	203,83
BIT 2,H	CB,54	203,84
BIT 2,L	CB,55	203,85
BIT 2,(HL)	CB,56	203,86
BIT 2,(IX+d)	DD,CB,d,56	221,203,d,86
BIT 2,(IY+d)	FD,CB,d,56	253,203,d,86
BIT 3,A	CB,5F	203,95
BIT 3,B	CB,58	203,88
BIT 3,C	CB,59	203,89
BIT 3,D	CB,5A	203,90
BIT 3,E	CB,5B	203,91
BIT 3,H	CB,5C	203,92
BIT 3,L	CB,5D	203,93
BIT 3,(HL)	CB,5E	203,94
BIT 3,(IX+d)	DD,CB,d,5E	221,203,d,94
BIT 3,(IY+d)	FD,CB,d,5E	253,203,d,94

INSTRUCCIONES DE PRUEBA DE BITS (II)

Código Fuente	Hexadecimal	Decimal
BIT 4,A	CB,67	203,103
BIT 4,B	CB,60	203,96
BIT 4,C	CB,61	203,97
BIT 4,D	CB,62	203,98
BIT 4,E	CB,63	203,99
BIT 4,H	CB,64	203,100
BIT 4,L	CB,65	203,101
BIT 4,(HL)	CB,66	203,102
BIT 4,(IX+d)	DD,CB,d,66	221,203,d,102
BIT 4,(IY+d)	FD,CB,d,66	253,203,d,102
BIT 5,A	CB,6F	203,111
BIT 5,B	CB,68	203,104
BIT 5,C	CB,69	203,105
BIT 5,D	CB,6A	203,106
BIT 5,E	CB,6B	203,107
BIT 5,H	CB,6C	203,108
BIT 5,L	CB,6D	203,109
BIT 5,(HL)	CB,6E	203,110
BIT 5,(IX+d)	DD,CB,d,6E	221,203,d,110
BIT 5,(IY+d)	FD,CB,d,6E	253,203,d,110
BIT 6,A	CB,77	203,119
BIT 6,B	CB,70	203,112
BIT 6,C	CB,71	203,113
BIT 6,D	CB,72	203,114
BIT 6,E	CB,73	203,115
BIT 6,H	CB,74	203,116
BIT 6,L	CB,75	203,117
BIT 6,(HL)	CB,76	203,118
BIT 6,(IX+d)	DD,CB,d,76	221,203,d,118
BIT 6,(IY+d)	FD,CB,d,76	253,203,d,118
BIT 7,A	CB,7F	203,127
BIT 7,B	CB,78	203,120
BIT 7,C	CB,79	203,121
BIT 7,D	CB,7A	203,122
BIT 7,E	CB,7B	203,123
BIT 7,H	CB,7C	203,124
BIT 7,L	CB,7D	203,125
BIT 7,(HL)	CB,7E	203,126
BIT 7,(IX+d)	DD,CB,d,7E	221,203,d,126
BIT 7,(IY+d)	FD,CB,d,7E	253,203,d,126

Fig. 10-1. Tabla de codificaciones para las instrucciones de prueba de bits.

SOLUCIONES A LOS EJERCICIOS

1.- La rutina podría ser:

```
190 IMPR_2 LD   B,8
200 BUCLE  LD   A,(DE)
210        LD   C,8
220 BUC_2  RR   A
230        RL   (HL)
240        DEC  C
250        JR   NZ,BUC_2
260        INC  DE
270        INC  H
280        DJNZ BUCLE
```

Entre las lineas 210 y 250 hemos introducido otro bucle que vá sacando los bits uno a uno por la derecha de "A" y metiéndolos uno a uno, tambien por la derecha, en el octeto correspondiente del archivo de pantalla.

2.- La única modificación necesaria es que la nueva dirección no habrá de almacenarse en "CHANS"+15, sino en "CHANS"+5. Por tanto, solo habrá que cambiar la linea 1010 para que sea:

```
1010       LD   DE,5
```

De paso, podemos modificar tambien la rutina "DESACT" cambiando, de la misma forma, la linea 1150.

3.- El procedimiento no puede ser más sencillo, basta con cargar en "DE" las nuevas coordenadas y hacer un salto a la etiqueta "SIGUE".

4.- Se almacenará, en el elemento #4 de la tabla, el dato "06" que corresponde al ofset del canal "S". La dirección correspondiente al elemento #4 de la tabla de corrientes es "STRMS"+14, es decir, 23568+14 = 23582. En la tabla de canales no se producirá ninguna modificación.

5.- La rutina podría ser:

```
100 MULT_2 LD   HL,0
110        LD   B,5
120 LOOP   SL   A
130        JR   NC,SIG
140        INC  H
150 SIG    DJNZ LOOP
160        LD   L,A
170        RET
```

Como se vé, realizamos cinco desplazamientos a la izquierda en el acumulador, incrementando "H" cada vez que sale un bit por el indicador de acarreo. Finalmente, cargamos en "L" el dato que haya quedado en "A".

GRUPO DE INSTRUCCIONES DE MANIPULACION DE BITS

Estas instrucciones actúan sobre la más elemental unidad de información, el bit. No es frecuente encontrar en los procesadores instrucciones dedicadas a manejar individualmente los bits, aunque supone una gran comodidad y operatividad el tenerlas.

Es de todos conocido la cantidad de alternativas binarias que existen a nivel informativo (si-no, blanco-negro, alto-bajo, hombre-mujer, derecha-izquierda, etc.). Pues toda esa información que sólo tiene dos posibilidades, o sea, que es binaria la podemos almacenar en un bit.

En cualquier procedimiento mecanizado se emplea mucho esta posibilidad, supone un gran ahorro de memoria en el almacenamiento de datos. Para analizar el estado de un bit utilizado como soporte de información se emplean muchos tipos de instrucciones, por ejemplo, los operadores lógicos con máscaras, o bien las instrucciones de desplazamiento llevando el bit a la posición de signo o de acarreo. Esas instrucciones y otras como de suma y resta se emplean para activar o desactivar un bit. Pues bien, todo esto se puede realizar directamente con las instrucciones que vamos a ver.

En este grupo de instrucciones existen tres subgrupos, a saber:

a) Prueba del bit (BIT): nos da la posibilidad de saber si un bit está activo o no.

b) Activar bit o puesta a uno (SET): pone a 1 (activo) un bit.

c) Desactivar bit o puesta a cero (RES): pone a 0 (desactivado o limpio) un bit.

El formato básico de estas instrucciones es el siguiente:

CODIGO b, OPERANDO

Donde «b» indica el bit sobre el que se va a operar. Los bits se numeran de derecha a izquierda, de 0 a 7.

7	6	5	4	3	2	1	0

El bit sobre el que se va a operar (valor de «b») viene indicado, a su vez, por tres bits del código de operación según la siguiente tabla:

Bit	valor binario de «b»
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Prueba de bits

BIT b,r

OBJETO:

Pone en el indicador de condición «Z» el complemento del valor del bit indicado por «b» en el registro indicado por «r». El código de representación de «r» es el indicado más abajo.

Registro	Código
B	000
C	001
D	010
E	011
H	100
L	101
A	111

CODIGO DE MAQUINA:

1 1 0 0 1 0 1 1
0 1 ← b → ← r →

INDICADORES DE CONDICION QUE AFECTA:

Z; pone 1 - si el bit especificado es 0;
pone 0 - en cualquier otro caso
H; pone 1 - siempre
N; pone 0 - siempre

Valor del octeto de memoria 8328h

8328:	0 1 1 0 1 0 0 1	69h

Instrucción

RES 5, (IX+4)	1 1 0 1 1 1 0 1	DDh
	1 1 0 0 1 0 1 1	CBh
	0 0 0 0 0 1 0 0	04h
	1 0 1 0 1 1 1 0	AEh

Valor del octeto de memoria 8328h después de la ejecución

8328:	0 1 0 0 1 0 0 1	49h

RES b, (IY + d)

OBJETO:

Pone a 0 el valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del registro índice «IY» más el entero de desplazamiento «d», el cual puede adquirir los valores desde —128 a + 127.

CODIGO DE MAQUINA:

1 1 1 1 1 1 0 1	FDh
1 1 0 0 1 0 1 1	CBh
← d →	
1 0 ← b → 1 1 0	

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

6

CICLOS DE RELOJ:

23

EJEMPLO:

RES 7, (IY–10)

Contenido del registro índice «IY»

(IY):	1 0 0 1 0 0 1 0	92h
	0 1 0 0 1 0 1 0	4Ah

Valor del octeto de memoria 9240h

9240h:	1 0 0 0 0 0 0 0	80h

Instrucción

RES 7, (IY–10)	1 1 1 1 1 1 0 1	FDh
	1 1 0 0 1 0 1 1	CBh
	1 1 1 1 0 1 1 0	F6h
	1 0 1 1 1 1 1 0	BEh

Valor del octeto de memoria 9240h después de la ejecución

9240h:	0 0 0 0 0 0 0 0	00h

Tablas de codificación

Dado el gran número de instrucciones que se utilizan para el manejo de bits, hemos realizado tres tablas de codificación. En la Figura 10-1 se encuentra la tabla con las instrucciones de prueba de bits, en la 10-2 la tabla con las de puesta a «1» y en la 10-3 las de puesta a «0». En la Figura 10-4 se encuentra la tabla resumida de indicadores y ciclos para todas estas instrucciones.

Los «Flags»

El término «Flag» se utiliza mucho en programación desde los primeros tiempos. Tiene su significado, al igual que otros términos, en el inglés; en el cual quiere decir bandera o banderín en los deportes, como verbo se traduce por hacer señales con banderas. En informática se empezó a utilizar para indicar que determinada condición, estaba puesta o no, también se utiliza la palabra «SWITCH», que significa interruptor.

La elección entre una palabra u otra (flag o switch) es más una cuestión de costumbre, aunque por regla general «flag» es un indicador de una condición y «switch» un cambio o bit que puede estar ON/OFF (conectado/desconectado). En ambos casos es una información binaria.

Los flags (banderas) son muy utilizados como condiciones por todos entendidas en la vida diaria. Por ejemplo una bandera roja en una playa indica que es peligroso bañarse; una serie de banderitas indican el camino a seguir en una pista de ski, etc.

También se da el uso de señales de tipo binario en otras aplicaciones, desde la protección de una cinta cassette a los semáforos. Todos ellos englobarían lo que en términos informáticos se entiende por flag.

Por tanto, flag sería una información binaria, que no tiene por qué ocupar más de un bit y que de cara a conseguir más efectividad debe ser fácil de poner, quitar y analizar.

Sería interminable enumerar aquí todas las aplicaciones que tienen los flags; de lo que se trata es de entender las posibilidades de su uso.

Uno de los usos más inmediatos es marcar las condiciones iniciales de un programa para posteriormente ir condicionando su ejecución.

74B2h:	0 1 1 0 0 1 1 1	67h

Puesta a «0» de bits

RES b, r

OBJETO:

Pone a 0 el bit indicado por «b» en el registro indicado por «r». El código de representación de «r» es el indicado más abajo.

Registro	Código
B	000
C	001
D	010
E	011
H	100
L	101
A	111

CODIGO DE MAQUINA:

1 1 0 0 1 0 1 1	CBh
1 0 ←b→ ←r→	

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

8

EJEMPLO:

RES 3,H

Valor del registro «H»

(H):	1 1 1 1 1 1 1 1	11h

Instrucción

RES 3,H:	1 1 0 0 1 0 1 1	CBh
	1 0 0 1 1 1 0 0	9Ch

Valor del registro «H» después de la ejecución

(H):	1 1 1 1 0 1 1 1	F7h

RES b, (HL)

OBJETO:

Pone a 0 el valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del par de registros «HL».

CODIGO DE MAQUINA:

1 1 0 0 1 0 1 1	CBh
1 0 ←b→ 1 1 0	

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

4

CICLOS DE RELOJ:

15

EJEMPLO:

RES 1, (HL)

Contenido del par de registros «HL»

(H):	1 0 0 0 0 0 1 0	82h
(L):	1 0 0 1 0 0 0 1	91h

Valor del octeto de memoria 8291h

8291h:	0 0 0 0 0 0 0 0	00h

Instrucción

RES 1, (HL):	1 1 0 0 1 0 1 1	CBh
	1 0 0 0 1 1 1 0	8Eh

Valor del octeto 8291h después de la ejecución

8291h:	0 0 0 0 0 0 0 0	00h

RES b, (IX + d)

OBJETO:

Pone a 0 el valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del registro índice «IX» más el entero de desplazamiento «d», el cual puede adquirir los valores desde —128 a + 127.

CODIGO DE MAQUINA:

1 1 0 1 1 1 0 1	DDh
1 1 0 0 1 0 1 1	CBh
←d→	
1 0 ←b→ 1 1 0	

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

6

CICLOS DE RELOJ:

23

EJEMPLO:

RES 5, (IX+4)

Contenido del registro índice «IX»

(IX):	1 0 0 0 0 0 1 1	83h
	0 0 1 0 0 1 0 0	24h

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

8

EJEMPLO:

BIT 3,C

Valor del registro «C»

(C):	0 1 1 1 0 1 0 1	75h

Instrucción

BIT 3,C:	1 1 0 0 1 0 1 1	CBh
	0 1 0 1 1 0 0 1	59h

El valor del registro «C» no varía con la ejecución

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	1	x	1	x	x	0	x

BIT b, (HL)

OBJETO:

Pone en el indicador de condición «Z» el complemento del valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del par de registros «HL».

CODIGO DE MAQUINA:

1 1 0 0 1 0 1 1	CBh
0 1 ← b → 1 1 0	

INDICADORES DE CONDICION QUE AFECTA:

Z; pone 1 - si el bit especificado es 0;
pone 0 - en cualquier otro caso
H; pone 1 - siempre
N; pone 0 - siempre

CICLOS DE MEMORIA:

3

CICLOS DE RELOJ:

12

EJEMPLO:

BIT 0, (HL)

Contenido del par de registros «HL»

(H):	1 0 0 1 0 0 1 1	93h
(L):	0 1 0 0 1 0 1 0	4Ah

Valor del octeto de memoria 934Ah

934Ah:	1 1 1 1 0 0 0 1	F1h

Instrucción

BIT 0, (HL):	1 1 0 0 1 0 1 1	CBh
	0 1 0 0 0 1 1 0	46h

El valor del octeto 934Ah no varía con la ejecución

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	0	x	1	x	x	0	x

BIT b, (IX + d)

OBJETO:

Pone en el indicador de condición «Z» el complemento del valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del registro índice «IX» más el entero de desplazamiento «d», el cual puede adquirir los valores desde —128 a + 127.

CODIGO DE MAQUINA:

1 1 0 1 1 1 0 1	DDh
1 1 0 0 1 0 1 1	CBh
← d →	
0 1 ← b → 1 1 0	

INDICADORES DE CONDICION QUE AFECTA:

Z; pone 1 - si el bit especificado es 0;
pone 0 - en cualquier otro caso
H; pone 1 - siempre
N; pone 0 - siempre

CICLOS DE MEMORIA:

5

CICLOS DE RELOJ:

20

EJEMPLO:

BIT 7, (IX+20)

Contenido del registro índice «IX»

(IX):	1 0 1 0 0 0 1 1	A3h
	1 0 1 1 0 0 1 0	B2H

Valor del octeto de memoria A3C6h

A3C6h:	0 1 1 1 1 1 1 1	7Fh

Instrucción

BIT 7, (IX+20):	1 1 0 1 1 1 0 1	DDh
	1 1 0 0 1 0 1 1	CBh
	0 0 0 1 0 1 0 0	14h
	0 1 1 1 1 1 1 0	79h

El valor del octeto de memoria A3C6H no varía con la ejecución

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	1	x	1	x	x	0	x

BIT b, (IY+d)

OBJETO:

Pone en el indicador de condición «Z» el complemento del valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del registro índice «IY» más el entero de desplazamiento «d», el cual puede adquirir los valores desde —128 a +127.

CODIGO DE MAQUINA:

```
11111101          FDh
11001011          CBh
   ← d →
01 ← b → 110
```

INDICADORES DE CONDICION QUE AFECTA:

Z; pone 1 - si el bit especificado es 0;
pone 0 - en cualquier otro caso
H; pone 1 - siempre
N; pone 0 - siempre

CICLOS DE MEMORIA:

5

CICLOS DE RELOJ:

20

EJEMPLO:

BIT 4, (IY−7)

Contenido del registro índice «IY»

```
(IY):  01111001   79H
       00011010   1Ah
```

Valor del octeto de memoria 7913h

```
7913h:  01011010   5Ah
```

Instrucción

```
BIT 4, (IY−7):  11111101   FDh
                11001011   CBh
                11111001   F9h
                01100110   61h
```

El valor del octeto de memoria 7913h no varía con la ejecución

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	0	x	1	x	x	0	x

Puesta a «1» de bits

SET b,r

OBJETO:

Pone a 1 el bit indicado por «b» en el registro indicado por «r». El código de representación de «r» es el indicado más abajo.

Registro	Código
B	000
C	001
D	010
E	011
H	100
L	101
A	111

CODIGO DE MAQUINA:

```
11001011          CBh
11 ← b → ← r →
```

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

8

EJEMPLO:

SET 2,A

Valor del registro «A»

```
(A):  00000000   00h
```

Instrucción

```
SET 2,A:  11001011   CBh
          11010111   D7H
```

Valor del registro «A» después de la ejecución

```
(A):  00000100   04h
```

SET b,(HL)

OBJETO:

Pone a 1 el valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del par de registros «HL».

CODIGO DE MAQUINA:

```
11001011          CBh
11 ← b → 110
```

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

4

CICLOS DE RELOJ:

15

EJEMPLO:

SET 5, (HL)

Contenido del par de registros «HL»

```
(H):  10010011   93h
(L):  01110100   74h
```

Valor del octeto de memoria 9374h

```
9374h:  10000110   86h
```

Instrucción

```
SET 5, (HL):  11001011   CBh
              11101110   EEh
```

Valor del octeto 9374h después de la ejecución

```
9374h:  10100110   A6h
```

SET b, (IX+d)

OBJETO:

Pone a 1 el valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del registro índice «IX» más el entero de desplazamiento «d», el cual puede adquirir los valores desde —128 a +127.

CODIGO DE MAQUINA:

```
11011101          DDh
11001011          CBh
   ← d →
11 ← b → 110
```

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

6

CICLOS DE RELOJ:

23

EJEMPLO:

SET 6, (IX+0)

Contenido del registro índice «IX»

```
(IX):  10110011   B3h
       01100100   64h
```

Valor del octeto de memoria B364h

```
B364h:  11111111   FFh
```

Instrucción

```
SET 6, (IX+0):  11011101   DDh
                11001011   CBh
                00000000   00h
                11110110   F6h
```

Valor del octeto de memoria B364h después de la ejecución

```
B364h:  11111111   FFh
```

SET b, (IY+d)

OBJETO:

Pone a 1 el valor del bit indicado por «b» en el octeto de memoria direccionado por el contenido del registro índice «IY» más el entero de desplazamiento «d», el cual puede adquirir los valores desde —128 a +127.

CODIGO DE MAQUINA:

```
11111101          FDh
11001011          CBh
   ← d →
11 ← b → 110
```

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

6

CICLOS DE RELOJ:

23

EJEMPLO:

SET 2, (IY−1)

Contenido del registro índice «IY»

```
(IY):  01110100   74h
       10110011   B3h
```

Valor del octeto de memoria 74B2h

```
74B2h:  01100011   63h
```

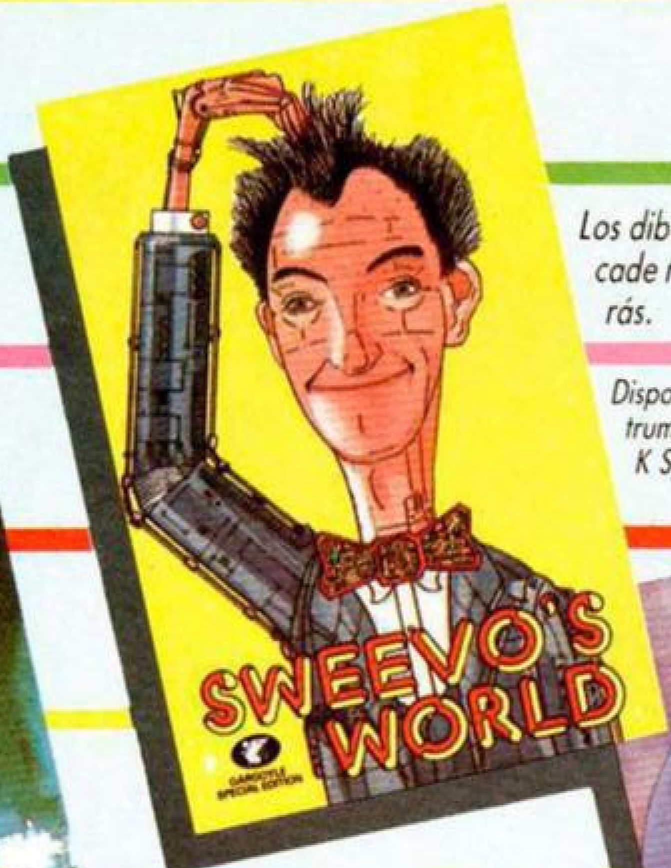
Instrucción

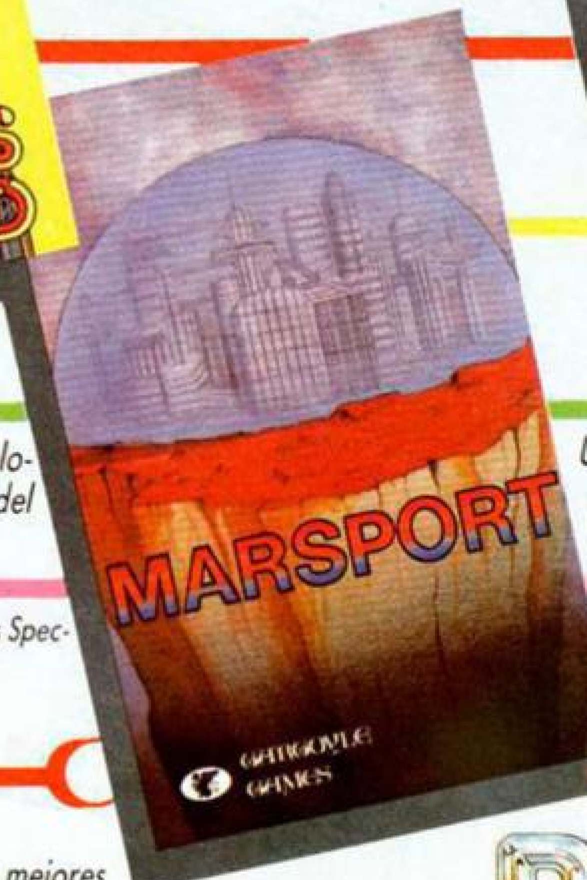
```
SET 2, (IY−1):  11111101   FDh
                11001011   CBh
                11111111   FFh
                11010110   D6h
```

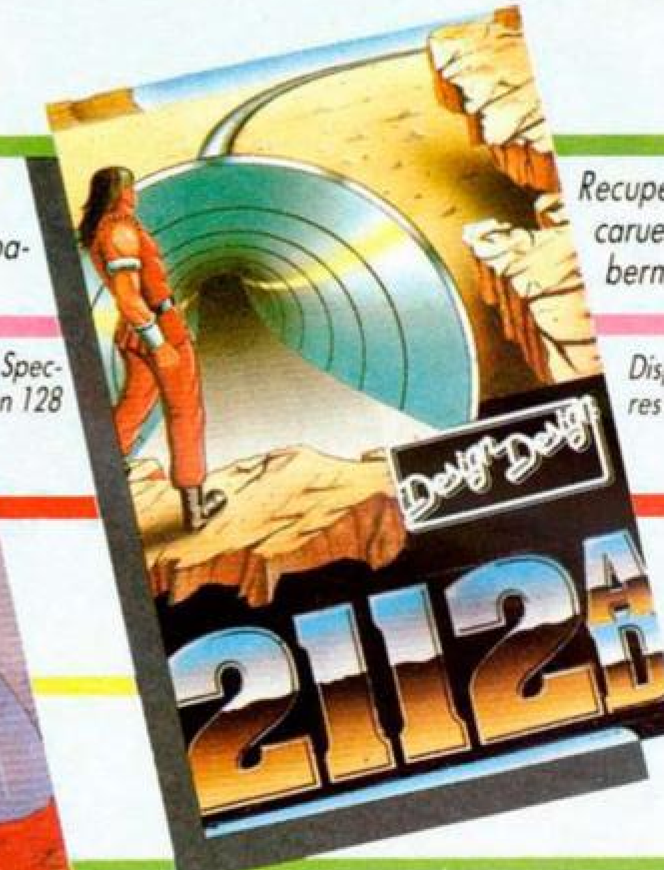
Valor del octeto de memoria 74B2h después de la ejecución

Nadie más que.....









Forbidden Planet

Arriesgar la vida y mucho más para localizar las partes del mensaje final del diabólico caballero.

Disponible para todos los ordenadores Spectrum y Amstrad.

Sweevo's World

Los dibujos animados de Arcade más cómicos en los que participarás.

Disponibles para todos los ordenadores Spectrum y Amstrad (y disco). Nueva versión 128 K Spectrum disponible ahora.

Marsport

Una aventura de Arcade de nuestros tiempos. Localice el ordenador central, y gane acceso al santuario exterior.

Disponible para todos los ordenadores Spectrum y Amstrad (y disco).

2112 AD

Recupera el control sobre el picaruelo ordenador que está gobernando el país.

Disponible para todos los ordenadores Spectrum Amstrad.

BUSCALO YA EN OnLine DE GALERIAS

Directamente disponibles desde BRITISH SOFT o en los mejores establecimientos de software.

Teléfono para tiendas, distribuidores y encargos contra reembolso: Tel. (965) 26 35 93. British Soft, S. A. Rocafel, 19. Albufereta (ALICANTE).

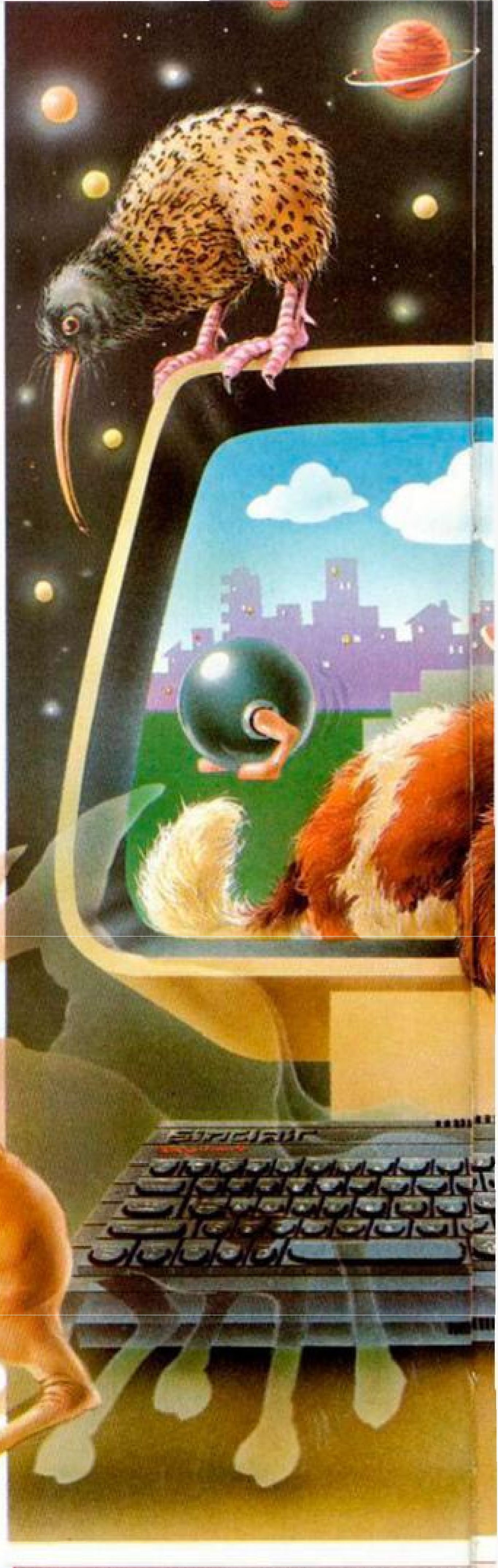
BRITISH SOFT

DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA

Distribuidores en Madrid: COMPUTIQUE.
C/. Embajadores, 90. 28012 Madrid
Tel. (91) 227 09 80

PROTAGONISTAS PARA TUS PROGRAMAS, DISEÑADOS POR DINAMIC

Las mentes calenturientas de los chicos de DINAMIC no descansan ni un solo instante en el desarrollo de nuevos programas, pero algunos de sus más afamados diseñadores de gráficos han hecho un alto en el camino ante la llamada de MICROHOBBY.



La mayoría de los programas que recibimos en nuestra redacción tienen un defecto común: Los gráficos de sus personajes son excesivamente simples, cuando no se mueven torpemente. Esta evidencia nos llevó, hace ya algunos meses, a la conclusión de que podía ser interesante convencer a diseñadores profesionales para que explicasen cuáles son sus técnicas. Aún más, lo ideal sería disponer de una serie de personajes con todas las facetas de su animación.

Tras arduas y prolongadas negociaciones, hemos conseguido que nada menos que los chicos de DINAMIC, se prestasen al experimento y creasen estos maravillosos personajes para que podáis utilizarlos en cualquiera de vuestros juegos.

Algunos son más pequeños que otros, tratando de dar facilidades aún a los menos hábiles, ya que cuanto más grandes, más difíciles de animar. También los hay con sólo dos movimientos, tres, cuatro, e incluso seis, que según los expertos, son las fases ideales.

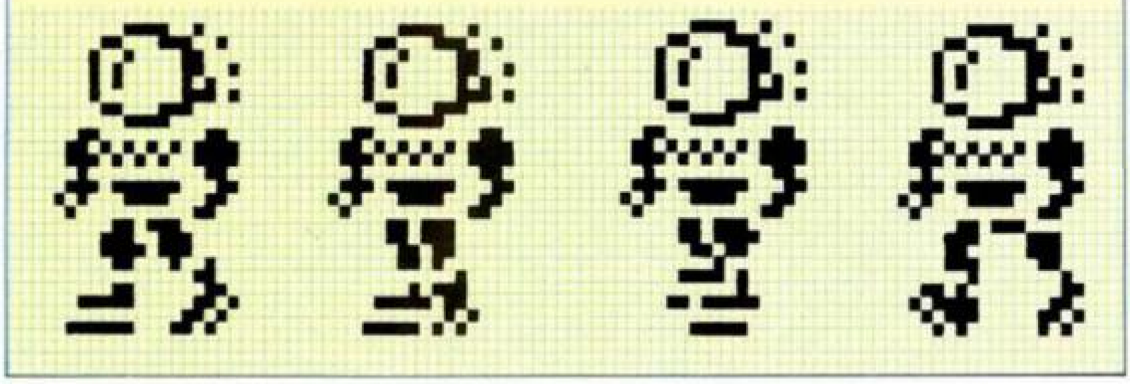
Con objeto de ocupar la mínima cantidad de memoria posible, los gráficos sólo están definidos en un sentido, de manera que cuando queramos que caminen en la otra dirección hay que invertir los gráficos con una pequeña rutina que también os ofrecemos. Por último, y sin que sirva de precedente, os damos también una rutina de movimiento, simple pero efectiva, que os servirá para haceros una idea de cómo se mueven estos personajes.

Ahora son vuestros protagonistas, tienen sus nombres e incluso sus historias. Podéis inventarles otras situaciones y aventuras y si os atrevéis a crear programas utilizando a todos o algunos de ellos, no dudéis en enviarnoslos. Pueden ser publicados en MICROHOBBY y llegado el caso, distribuidos por DINAMIC en su sello FUTURE STARS. Esperamos que os animéis a dar vida a estos simpáticos personajes.

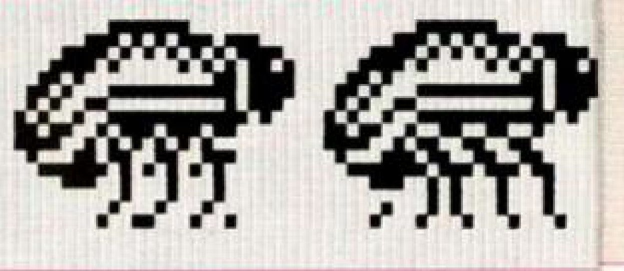
MANOLO MINGLANILLAS

Astronauta. El primer albaceteño que pisó la superficie lunar. Cursó estudios de Ingeniería Aeronáutica en la Universidad de Houston (Cuenca), pero nunca llegó a finalizarlos. Su primer viaje al espacio lo realizó al despegar accidentalmente una nave en la que realizaba servicios de limpieza y mantenimiento.

—Cagonlá, la que sa liao aquí!, fueron sus primeras declaraciones desde el espacio sideral recogidas desde la base de seguimiento de Fresnedillas.



«CUCA», LA CUCARACHA





Totalmente vilipendiada, el ortóptero nocturno y corredor no cesa de esconderse entre humedades y porquerías pensando en la mejor manera de sobrellevar su carga. En un último intento desesperado ha pensado enrrolarse en la marina creyendo, la muy ilusa, que con los aires marinos y el balanceo podrá desechar su fama. ¿Lo conseguirá?...

PROGRAMA BASIC. Demostración

```
  10 BORDER 0: PAPER 0: INK 7: C
LEAR 49999
  20 LOAD ""CODE 6E4: LOAD ""COD
E 5E4: LOAD ""CODE 61000
  30 RANDOMIZE USR 61012: POKE 2
3658,8: RANDOMIZE USR 61000
  40 CLS : PRINT AT 5,11;"* M E
N U *"
  50 PRINT AT 8,13;"1. KIUI"
  60 PRINT AT 10,10;"2. ASTRONAU
TA"
  70 PRINT AT 12,10;"3. CUCARACH
A"
  80 PRINT AT 14,13;"4. BOLA"
  90 PRINT AT 16,12;"5. SALTA"
 100 PRINT AT 18,12;"6. PERRO"
 110 LET T$=INKEY$
 120 IF CODE T$<49 OR CODE T$>54
 THEN GO TO 110
 130 LET N=1: LET SU=0: LET X=10
: LET Y=14
 150 GO TO (VAL T$*1000)
1000 REM MOVIMIENTO KIUI
1010 POKE 60001,3: POKE 60002,2
1020 DIM A(6)
1030 LET A(1)=50000: LET A(2)=50
048
1040 LET A(3)=50096: LET A(4)=50
144
1050 LET A(5)=50192: LET A(6)=50
240
1060 LET L=6
1070 CLS
1990 GO TO 7070
2000 REM MOVIMIENTO ASTRONAUTA
2010 POKE 60001,2: POKE 60002,3
2020 DIM A(4)
2030 LET A(1)=50288: LET A(2)=50
336
2040 LET A(3)=50384: LET A(4)=50
432
2050 LET L=4
2060 CLS
2990 GO TO 7070
3000 REM MOVIMIENTO CUCARACHA
3010 POKE 60001,3: POKE 60002,2
3020 DIM A(2)
3030 LET A(1)=50480: LET A(2)=50
528
3040 LET L=2
3050 CLS
3990 GO TO 7070
4000 REM MOVIMIENTO BOLA
4010 POKE 60001,2: POKE 60002,2
4020 DIM A(6)
4030 LET A(1)=50576: LET A(2)=50
608
4040 LET A(3)=50640: LET A(4)=50
672
4050 LET A(5)=50704: LET A(6)=50
736
4060 LET L=6
4070 CLS
4990 GO TO 7070
5000 REM MOVIMIENTO SALTA
5010 POKE 60001,2: POKE 60002,3
5020 DIM A(3)
5030 LET A(1)=50768: LET A(2)=50
816
5040 LET A(3)=50862
5050 LET L=3
5060 CLS
5990 GO TO 7070
6000 REM MOVIMIENTO PERRO
6010 POKE 60001,2: POKE 60002,2
6020 DIM A(4)
6030 LET A(1)=50910: LET A(2)=50
946
6040 LET A(3)=50976: LET A(4)=51
008
6050 LET L=4
6060 CLS
6990 GO TO 7070
7000 REM MOVIMIENTO DER.-IZQ.
7010 LET T$=INKEY$
7020 IF CODE T$<79 OR CODE T$>81
 THEN GO TO 7010
7030 IF CODE T$=81 THEN RANDOMIZ
E USR 61000: GO TO 40
7040 RANDOMIZE A(N): PRINT AT X,
Y;: RANDOMIZE USR 6E4
7050 LET N=N+1: IF N>L THEN LET
N=1
7055 IF SU=1 AND CODE T$=79 THEN
 RANDOMIZE USR 61000
7056 IF SU=0 AND CODE T$=80 THEN
 RANDOMIZE USR 61012: LET SU=1
7060 IF CODE T$=79 AND Y>=1 THEN
 LET SU=0: LET Y=Y-1
7065 IF CODE T$=80 AND Y<=28 THE
N LET SU=1: LET Y=Y+1
7070 RANDOMIZE A(N): PRINT AT X,
Y;: RANDOMIZE USR 6E4
7075 FOR A=1 TO 20: NEXT A
7100 GO TO 7010
```

Teclear y salvar en cinta con SAVE «DEMO» LINE 1.

RUTINA DE VOLCADO EN PANTALLA

Linea	Datos	Control
1	010202ED43BBEAAF2121	971
2	18ED5B885CED52EB2A76	1294
3	5C7932BDEA7882F5F13D	1483
4	BAD83CF5D57AE618C640	1558
5	477A0F0F0FE6E0835F50	998
6	06083ABDEA4FD5E579FE	1391
7	0028081AAE1213230D18	357
8	F3E1ED5BBBEA160019D1	1473
9	1410E1D1B7DCBFEA1418	1342
10	C1020202000000000000	199

Usando el Cargador Universal, teclea y salva este bloque en 2.° lugar. DUMP 40000. N° de bytes 94.

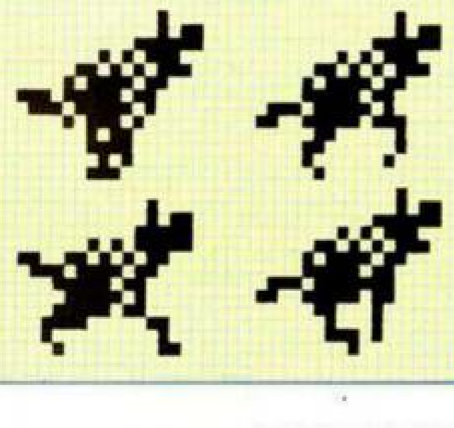
PIWI KIWI

Un silencioso «pájaro» que deambula tímidamente picoteando todo lo que encuentra a su paso (lo que le ha supuesto alguna que otra bronca por poner el pico donde no debe). Sobre su historia poco se sabe, tan sólo que su fama de picoteador sobrepasa los confines de su barrio. Por algo será.



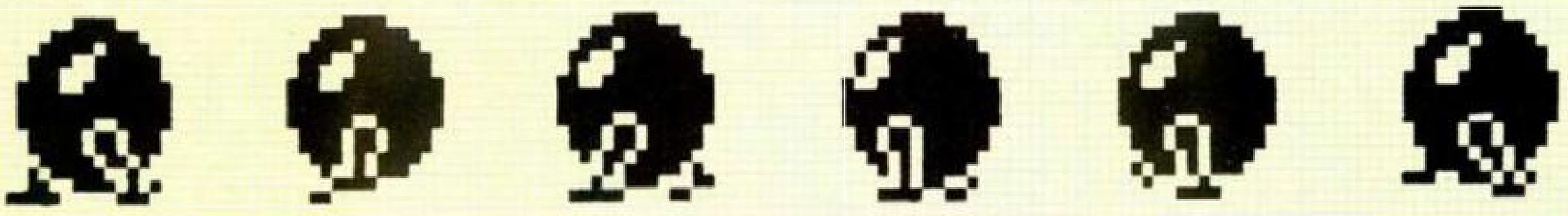
«PULGAS», EL PERRO

Este lanudo y desaliñado personaje no es lo que parece. «Pulgas», como se le conoce en el mundo del hampa, no es sino un infiltrado policía cuya intención es la de desarticular la más importante red de gánster desde los tiempos de Al Capone. ¿Será suficiente este espectacular disfraz para burlar la dura mano de la mafia...?



RODOB EL SOLOB

Simpática bolita especialmente habilitada, gracias a sus dos cortas pero potentes patas, para practicar cualquier tipo de deporte. Por un sofisticado sistema de autopropulsión, la velocidad media que puede llegar a alcanzar, puede tener diversas aplicaciones. Es cuestión de echarle imaginación.





«MARCHOSO» CANGOSO

¿Será un canguro, será una flor...? Su ambigua personalidad crea especulaciones a su paso. A este respecto, Radio Macuto ha asegurado que este marchoso personaje es en realidad un príncipe encantado a la búsqueda y captura de una princesa que le libere. Esta prestigiosa emisora no ha sabido exactamente de qué quiere liberarse nuestro personaje.



DATOS DE LOS GRAFICOS

Linea	Datos	Control
1	001D00006480000B34000	500
2	DEC01DCA603FF9A03FD7	1491
3	802FF2E03FFE601DFCC0	1527
4	107B40203B00202D8040	563
5	2040404050000E1E0001D	782
6	00006480000B34000DEC0	885
7	01CA601DF9A03FD7803F	1206
8	F2E02FFE603FF9C01C76	1513
9	40103680201500200700	354
10	400880401DC0001D0000	514
11	648000B34000DEC01DCA	1116
12	603FF9A03FD7802FF2E0	1487
13	3FFE601DF3C0106D4020	1098
14	2DA0201B204010104020	488
15	1C00F860001D00006480	629
16	00B34000DEC001CA601D	985
17	F9A03FD7803FF2E02FFE	1645
18	603FF3C01C6D40101D80	968
19	200B40200084040086040	443
20	3D80001D00006480000B3	625
21	4000DEC01DCA603FF9A0	1277
22	3FD7802FF2E03FFE601D	1361
23	F3C0106D40202D802015	882
24	00407400400200000F00	261
25	001D00006480000B34000	500
26	DEC001CA601DF9A03FD7	1429
27	803FF2E02FFE603FFBC0	1560
28	1C754010368020180020	498
29	210040408041E1C003C8	974
30	0C721218141914141010	298
31	0C7003C030066AAF754F	850
32	200677E3A3C6400C06E0	1051
33	0E700F700624000C0606	319
34	3E0D001A7E3403C80C72	608
35	121814191414101D0C70	296
36	03C030066AAF754F2006	764
37	77E3A3C6400C06E006E0	1243
38	03600340001001700FB0	486
39	00281F5003C80C721218	522
40	141914141 01D0C7003C0	449
41	30066AAF754F200677E3	915
42	A3C6400C06E006F00360	1012
43	008007A000200DF00000	580
44	03E003C80C7212181419	643
45	1414101D0C7003C03006	458
46	6AAF754F200677E3A3C6	1222
47	400C06E00E1C0C1C0E0C	414
48	020234035E062E0D140A	248
49	007F001DD5C07AAAF0B5	1274
50	4178F6002CF7FF167400	1115
51	CA2FFF2504CC9B03334F	1037
52	0666E20DDDFC088A7008	1086
53	8A00088A2011111110007F	493
54	001DD5C07AAAF0B54178	1332
55	F6002CF7FF167400CA2F	1179
56	FF2506EE9B01554F0733	914
57	6202AAFC022270022200	706
58	01110004 8C80000003C0	485
59	0FF01F783CFC38FC79FE	1401
60	7FFE7FFE7F3E3EDC3EDC	1515
61	1F682F9033D4F8380000	893
62	03C00FF01EF839FC31FC	1338
63	73FE7FFE7FFE7E7E3DBC	1632
64	3DBC1D78097007801800	678
65	000003C00FF01DF833FC	1030
66	23FC67FE7FFE7FFE7CFE	1784
67	3B7C3B7C16F805F409C6	1092
68	3E18000003C00FF01BF8	811
69	27FC27FC4FFE7FFE7FFE	1677
70	79FE36FC36FC1AF80AF0	1511
71	02C80F30000003C00FF0	715
72	1DF833FC23FC67FE7FFE	1605
73	7FFE7CFE3B7C3B7C1D78	1274
74	2D7010800 3C0000003C0	691
75	0FF01EF839FC31FC73FE	1512
76	7FFE7FFE7E7E3DBC3DBC	1512
77	1ED81F5073A800700000	752
78	000000003E00DF83FCC	755
79	60828100010003800 7C0	686
80	05E07BE047B01F581B9C	1125
81	0DDA0DAE066401B0001C	729
82	000C001C01E000000000	265
83	03E00DF83FCC60828100	1110
84	010003800074075A04BE0	779
85	07B01F5C1B9A0DDE0DA4	899
86	066001B8000C000C000C	323
87	0030000003E00DF83FCC	803
88	60828100010003800740	558
89	05A07BE047B01F581B9C	1061
90	0DDA0DAE066401B8000C	721
91	00080008001000100000	48
92	00000000000000001000	16
93	D000FC003A8075601BD0	1094
94	0DF00FEE0DF304600440	930
95	05C00100000000000000	198
96	00001000D000F800FA80	850
97	35661BDC0DF00BE00DF0	1143
98	30782008601000000000	320
99	00001000D000FC003A80	662
100	75601BD00DF00BEE1DF3	1222
101	10601030081000200000	232
102	00001000D000FC003A80	662
103	75601BD00DF70BEE05D0	1170
104	034003C002800 1C00000	585

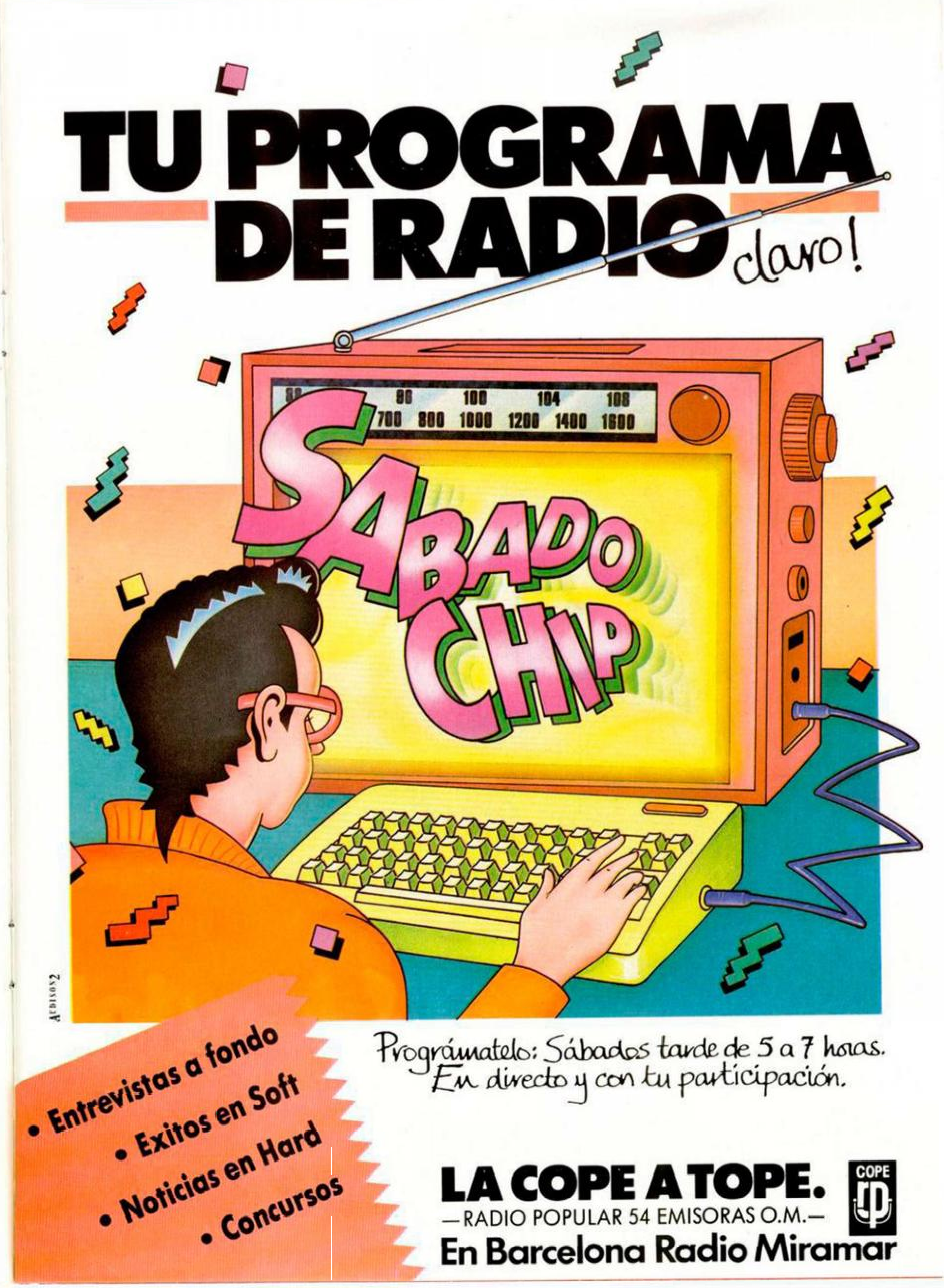
Con el Cargador Universal salvar este bloque en 3er lugar. DUMP 40000. N.° de bytes 1040.



RUTINA DE INVERSION DE GRAFICOS

Linea	Datos	Control
1	2170CB1150C3011004ED	898
2	B0C92150C31170CB0110	1034
3	04EDB02150C322ADEE21	1203
4	60C722AFEEDD21FBEE06	1491
5	19C5DD7E0032F9EEDD23	1362
6	DD7E0032FAEECDB1EEDD	1726
7	23DD7E00DD232AAFEE16	1115
8	005F1922AFEE2AADEE19	1045
9	22ADEEC110D3CDDEEE21	1563
10	60C71150C3011004EDB0	1021
11	C960C770CB2AADEE3AF9	1571
12	EE16005F192BED5BAFEE	1164
13	3AFAEE47C5E53AF9EE47	1659
14	7E12132B10FAE1D53AF9	1217
15	EE16005F19D1C110E7C9	1230
16	2160C71160C7011004C5	858
17	1A06081FCB1610FB1323	617
18	C10B78B120EFC9021003	994
19	103003103000310300310	217
20	300310300031030021830	256
21	02183002183002183003	225
22	103003103000210200210	199
23	200210200021020021020	182
24	02102002183002183002	200
25	183002102000210200210	190
26	200210 20C90000000000	283

Salvar en la cinta, en 4.° lugar este bloque, indicado DUMP 40000. N.° de bytes 255.


TU PROGRAMA DE RADIO
claro!
88 90 100 104 108
700 800 1000 1200 1400 1600
SABADO CHIP
• Entrevistas a fondo
• Exitos en Soft
• Noticias en Hard
• Concursos
Prográmatelo: Sábados tarde de 5 a 7 horas.
En directo y con tu participación.
LA COPE A TOPE.
—RADIO POPULAR 54 EMISORAS O.M.—
En Barcelona Radio Miramar
COPE
rp

Microfile

MAPA DE SECTORES DEL MICRODRIVE

Luis E. JUAN VIDALES

En la operación de formateo, el microdrive se divide en un determinado número de sectores de los que algunos quedan defectuosos. Posteriormente, a medida que salvamos programas, los sectores libres se van ocupando. Con esta práctica rutina podéis examinar vuestros cartuchos averiguando los sectores defectuosos, libres y ocupados.

Tal como se indicaba en la sección MICROFILE correspondiente al n.º 50 de la revista, en la operación de formateado del cartucho se crean unos 200 sectores capaces de almacenar 512 bytes cada uno. Más concretamente, las operaciones que tienen lugar en el formateado son las siguientes:

a) Se graban 254 sectores, numerados de 254 a 1, conteniendo todos ellos un mismo bloque de datos patrón. En realidad, debido a la longitud del bucle de cinta contenido en el cartucho, los aproximadamente 70 últimos sectores se graban encima de los primeros. De esta curiosa manera, todos los sectores resultan numerados correlativamente en modo decreciente, terminando siempre en el n.º 1, independientemente de la longitud de la cinta.

b) Se leen los sectores grabados, calculando sus checksum.

c) Se marcan como sectores libres aquéllos cuyos checksum son los correctos. Los defectuosos quedan grabados con el patrón inicial y son inutilizables posteriormente.

La rutina que publicamos esta semana visualiza en tiempo real el mapa del cartucho introducido en el microdrive 1, indicando el número de sectores libres, usados y defectuosos, así como su posición relativa en la cinta. Mediante ella, podremos observar, por ejemplo, cómo se produce el llenado del cartucho desde la operación de formateado, cada vez que se almacena un fichero nuevo. En cada momento, el número de K disponibles es la mitad del n.º de sectores libres (1K = 1024 = 2×512).

Como siempre, para averiguar el tipo de ROM contenido en nuestra Interface 1, ejecutaremos el comando:

CLOSE#0 : PRINT PEEK 23729

El resultado nos indicará:

80 = ROM tipo 1

0 = ROM tipo 2

La rutina tiene su origen en la dirección 64500 y consta de 249 bytes. Introducido el listado correspondiente con el consabido cargador de C/M, la rutina se ejecutará con:

RANDOMIZE USR 64500

ROM TIPO 1

```
 1  CF31D9E5D93E02CD0116  1211
 2  21000022F6FC22F8FC22  1133
 3  FAFC2112FC22ED5CCF32  1425
 4  E1E1CDA5103E01CD3215  1175
 5  CDA91321EDFC110900CD  1146
 6  A9FC21EDFCCB4620ED3A  1543
 7  FAFCB72008DD7E2932FA  1413
 8  FC1805DDBE29283021F9  1103
 9  FC343AEDFC21F0FCB6E6  1788
10  02200021F6FC343E2E18  757
11  143AF1FCB7280821F7FC  1334
12  343E2A180621F8FC343E  833
13  8FCD711D18AACDADFCCD  1519
14  ADFCDDE5AFCD32150E04  1344
15  21BDFCDD21F6FC060C7E  1370
16  CD711D2310F9DD7E00CD  1199
17  B2FCCDADFCDD230D20E9  1594
18  DDE1CD9F11CD0007E1D9  1481
19  C9E5C3F2153E00C37110  1300
20  C5E5D7282DD7E32DE1C1  1631
21  C92E204C494252455320  760
22  203A202A204F43555041  572
23  444F533A208F20444546  702
24  4543542E203A20544F54  635
25  414C20534543543A2000  566
```

ROM TIPO 2

```
 1  CF31D9E5D93E02CD0116  1211
 2  21000022F6FC22F8FC22  1133
 3  FAFC2112FC22ED5CCF32  1425
 4  E1E1CDE60F3E01CDF717  1440
 5  CDC41221EDFC110900CD  1172
 6  A9FC21EDFCCB4620ED3A  1543
 7  FAFCB72008DD7E2932FA  1413
 8  FC1805DDBE29283021F9  1103
 9  FC343AEDFC21F0FCB6E6  1788
10  02200021F6FC343E2E18  757
11  143AF1FCB7280821F7FC  1334
12  343E2A180621F8FC343E  833
13  8FCD661D18AACDADFCCD  1508
14  ADFCDDE5AFCDF7170E04  1543
15  21BDFCDD21F6FC060C7E  1370
16  CD661D2310F9DD7E00CD  1188
17  B2FCCDADFCDD230D20E9  1594
18  DDE1CDC410CD0007E1D9  1517
19  C9E5C3AD183E00C36610  1223
20  C5E5D7282DD7E32DE1C1  1631
21  C92E204C494252455320  760
22  203A202A204F43555041  572
23  444F533A208F20444546  702
24  4543542E203A20544F54  635
25  414C20534543543A2000  566
```



MDS (Microdrive disk system) de PIN-SOFT

José Manuel LAZO

Bajo este pomposo nombre se esconde un paquete de utilidad que amplía el repertorio de comandos del microdrive con 6 nuevos que sirven, fundamentalmente, para permitir un acceso más racional a la información que depositemos en nuestros cartuchos.

El sistema se presenta en un cartucho de microdrive en el cual está grabado un número elevado de veces para evitar que lo perdamos por un error de carga. Después de esperar un corto tiempo mientras el paquete se instala en nuestro ordenador, dispondremos ya de los nuevos 6 comandos que abajo se comentan.

Los microdrives de Sinclair tienen una particularidad interesante, y es la posibilidad de almacenar datos en un cartucho abriendo previamente una «corriente» asignada al mismo, lo que es bueno a la hora de confeccionar ficheros y utilidades de gestión parecidos. Pero tiene el inconveniente de que una vez que hayamos cerrado el fichero (CLOSE) no podremos modificarlo, ni borrar una serie de datos específicos, ni cambiarlos por otros, ni ampliar el archivo, teniendo que abrir otro paralelo y copiar el contenido del original en el de destino con los datos modificados.

Asimismo, el acceso a los datos de esta forma ha de ser forzosamente secuencial, es decir, empezar por el principio hasta llegar a la ficha en cuestión que deseemos.

Pues bien, el sistema MDS es, básicamente, lo mismo, pero sin todos los defectos que arriba se han puesto. Se puede abrir un fichero de una longitud determinada a priori, y modificar los datos de ese fichero de la forma que más nos plazca, y además, se dispone de órdenes para situar la cinta del cartucho directamente sobre un dato específico, sin tener que pasar primeramente sobre los anteriores. Asimismo, el manejo de la información es mucho más racional, optimizándose en gran medida el tiempo de espera para las operaciones de escritura y lectura.

Los comandos

Es interesante dar un vistazo sobre los nuevos comandos que se ofrecen, aunque en el manual que acompaña al programa se expone la forma de manejarlos de una manera clara y concisa.

La orden RND FORMAT sirve para FORMATear o DIMensionar un determinado espacio en la cinta del cartucho, esta es la primera operación que debemos hacer cuando vayamos a usar un fichero por crear.

Así, si deseáramos reservar en el cartucho 50K un fichero llamado «clientes», deberíamos teclear en modo directo, o en modo programa: RND FORMAT «clientes»; 1; 50. El uno significa que vamos a usar el microdrive nº 1.

Una vez creado el fichero podemos disponer ya de él, bien de forma aleatoria, con los comandos que abajo se exponen o de forma secuencial, con los comandos normales del microdrive. Su nombre aparecerá en catálogo como si de un programa normal se tratase.

Para usarlo debemos primero abrirlo, pero sin necesidad de tenerlo que asignar a ninguna corriente específica, ya que sólo se puede tener abierto un fichero aleatorio a la vez.

La orden para hacer esto es RND OPEN # teniendo la siguiente sintaxis para nuestro ejemplo: RND OPEN «Clientes»; 1; 30. Esto significa: abrir el archivo «clientes» disponiendo de 30 octetos cada ficha.

Esto último tiene una serie de particularidades muy especiales: Si cada ficha dispone de 2 campos, por ejemplo, y estos campos están determinados por dos variables alfanuméricas, tendremos que determinar la longuitud de la ficha por la suma de las longitudes de las dos variables, más un octeto por cada variable que se utilice. Así, si en nuestro ejemplo hemos determinado 30 octetos por ficha, ello puede corresponder a dos variables, A$ y B$, con longuitud 10 y 18, por ejemplo.

Una vez abierto el fichero y antes de hacer cualquier operación de escritura o lectura, deberemos de apuntar a una ficha determinada. Esto se hace con el comando POINT AT ficha. Una vez ejecutado el sistema determinará si ha de hacer alguna operación de microdrive para apuntar a la ficha.

Cuando hayamos apuntado a la ficha que deseemos ya sólo queda escribir en ella o leer de la misma. Para cualquiera de estas dos operaciones se ha de DIMensionar previamente una variable alfanumérica con la longitud adecuada ya que los comandos de lectura y escritura sólo funcionarán si la variable tiene definida su longitud.

Así, el comando de escritura es RND PRINT variable, y en nuestro supuesto ejemplo sería RND PRINT A$: RND PRINT B$ teniendo en ambas variables el texto de los campos que vayamos a utilizar.

De igual manera el comando de lectura de una ficha es RND INPUT variable utilizando una sintaxis idéntica a la orden de escritura.

Si se han de hacer varias operaciones de lectura y/o escritura sobre una misma ficha es necesario apuntar cada vez sobre la misma con el comando POINT AT.

El último comando, ya como colofón, es RND CLOSE# que, como ya habreis podido imaginar, cierra el fichero aleatorio que estemos usando, esto es muy importante antes de terminar el trabajo para efectuar una escritura física en la cinta del cartucho del último bloque de fichas que hayamos manejado.

Este fichero aleatorio lo podremos tener abierto a la vez de la forma normal y con los comandos especiales sin haberse encontrado ningún problema en ello.

Resumiendo: con el nombre de «MDS» adquirimos un paquete con estos seis comandos que nos ayudan notablemente en el manejo de ficheros en el microdrive de Sinclair.

TODOS LOS MESES

Regalamos una moto o un radiocassette







¡No te pierdas este número!

Recorta o copia este cupón y envíalo a Hobby Press, S.A. Apartado de Correos 232. Alcobendas (Madrid)

Nombre ______ Apellidos ______

Dirección ______ C. Postal ______

Localidad ______ Provincia ______

Teléfono ______ Profesión ______

☐ *Deseo suscribirme a* ***MICROMANIA*** *por un año (11 números) al precio de 2.850 pesetas. (IVA incluido).*

Forma de pago. Marca con una X la opción que desees.

☐ Contra reembolso.

☐ Mediante tarjeta de crédito VISA. Número de la tarjeta ______

Fecha caducidad de la tarjeta ______

☐ Mediante talón bancario a nombre de Hobby Press, S.A.

☐ Mediante giro postal n.º ______

☐ Mediante domiciliación bancaria

Banco ______ Sucursal y Localidad ______

N.º de cuenta ______ Fecha y Firma ______

Titular de la cuenta ______

MICRO Manía

Sólo para adictos

THREE WEEKS IN PARADISE

Parecía que ya estaba todo dicho a cerca de este fenomenal **Three Weeks in Paradise**, pero Angel Martínez Castaño nos ha demostrado desde Madrid que esto no era verdad, pues nos ha enviado un truco de lo más interesante. Consiste en lo siguiente:

— Ir al pozo e introducirse en su interior.

— Una vez abajo, pulsar simultáneamente «P», «D» y «Symbol Shift». Sonará una música indicando que tenemos vidas infinitas.

— Salir del pozo y comenzar a jugar como si tal cosa.

CAMELOT WARRIORS

En el número 70 de vuestra revista favorita aparecía un poke para obtener varias vidas infinitas y facilitaros las cosas en este difícil arcade. Sin embargo, en vista de que hemos recibido algunas llamadas y cartas quejándose al respecto de que dicho poke no funcionaba, cosa ésta bastante posible sin tenemos en cuenta la gran cantidad de copias piratas diferentes que por desgracia circulan por todas partes, os vamos a ofrecer el único, el genuino, el original e inigualable poke de vidas infinitas para el Camelot Warrios, el cual ha sido gentilmente remitido por los mismísimos programadores de Dinamic.

POKE 50783, 200

Y si además queréis que no os acosen los bichos, no tenéis más que teclear lo que a continuación se suscribe:

POKE 55918, 201

(Como siempre, antes de Randomize USR).

SPIDERMAN

En esta ocasión seremos escuetos e iremos al grano, que queda poco espacio y muchas cosas que contar.

1.° Para matar a Hidroman: subir por la caja del ascensor hasta que no podáis más y pulsar «Push Elevator», con lo que subiremos un piso má arriba. Salir de la caja del ascensor y veréis un termostato. Escribimos «Lower» y bajamos a la planta de abajo: Hidroman se habrá congelado. Cogemos el bloque de hielo y subimos de nuevo. Volvemos a subir el termostato y el hielo se derretirá. Vaciamos el acuarium y cogemos la gema que se encuentra en su interior.

2.° Para acabar con Lizard: vamos al laboratorio y cogemos los potingues de la oficina. Dejamos las «Chemicals» y entramos en el laboratorio. Ponemos «Make Fórmula» y conseguiremos un líquido que, al dejarlo delante de Lizard, hará que éste se convierta en el Doctor Conners.

3.° Para evitar a Sandman no tendremos más que subirnos al techo y mirar en su crib, lo que te dará acceso a coger una interesante fórmula que te será de futura utilidad.

Esto se lo debéis a Felipe Bertrand, quien de antemano ya os devuelve el agradecimiento.

LIBROS



COMO HACER ROBOTS CONTROLADOS POR ORDENADOR

Tony Potter y Chris Oxlade

Ediciones Generales Anaya

48 páginas

Un robot es una máquina controlada por ordenador, programada para manejar herramientas o mover objetos. En este libro pueden aprenderse algunos fundamentos de la Robótica y abordar con bastantes posibilidades de éxito la construcción de un Robot gobernado por nuestro Spectrum.

El libro está dividido en secciones, dedicadas cada una de ellas a la construcción y verificación de una parte del robot.

Se tratan también algunos conceptos básicos de electrónica, siempre enfocados a la práctica, teniendo en cuenta que una de las fases es la realización del circuito de control del robot.

En un principio puede ser necesario seguir las indicaciones del libro al pie de la letra, sobre todo para los no iniciados. Sin embargo, a medida que se adquiere la suficiente práctica se está más capacitado para crear nuestras propias variaciones del modelo de Robot que aquí se presenta.

Se trata de un micro-robot de brazo capaz de coger cosas y trasladarse de un sitio a otro mediante un par de ruedas.

Además de la descripción de todos los pasos, su construcción contiene dibujos a tamaño real de las plantillas que hay que recortar para la elaboración de las piezas.

Incluye también una completísima lista de todos los materiales que van a ser necesarios para realizar tan apasionante tarea, así como un programa en Basic que se encargará de gestionar todos los movimientos de nuestro robot.

Lo más destacable es la gran cantidad de consejos prácticos y el desarrollo, paso a paso, de todas las facetas del montaje, lo que permite una rápida asimilación por usuarios de todas las edades.

En definitiva, un libro altamente recomendable para los manitas que tendrán sobradas oportunidades de demostrar sus habilidades con el lápiz, el soldador, el destornillador, la brocha y finalmente, el teclado de su ordenador. Todo un reto.

Cómo entrar en un programa y averiguar sus secretos

LA BIBLIA DEL «HACKER» (V)

José Manuel LAZO

La semana pasada analizábamos la utilidad de un programa, viejo conocido nuestro, COPYLINE, en las tareas de análisis de los cargadores Basic. Ahora continuaremos con esta labor incluyendo una interesante tabla que recoge todos los controles de color y cursor que maneja el Spectrum.

El programa en cuestión nos lista un Basic que esté ubicado en otra dirección aunque tenga cualquier protección de controles de color o cursor. El listado lo produce en 5 columnas cuyo significado se explicó la pasada semana.

En la antepenúltima columna van los controles de color, cursor, etc. Pero éstos no actúan sobre el listado. Consultando la tabla adjunta puedes averiguar la función de cada uno.

Estos últimos controles que son a modo de prefijos para los parámetros que le acompañan, con unos argumentos erróneos, hacen que el SO se confunda bastante a la hora de sacar el listado.

Con el Copyline tenemos, además, la ventaja de que al no modificar ninguna de las partes del programa y no estar éste en la zona del Basic no se corrompen la zona de las variables ni la zona de edición, lugar en el que se pueden volcar programas en CM tal y como veremos próximamente.

Vamos a tratar ahora de la protección que raya la frontera entre el Basic y el CM. Es el caso de los cargadores que tengan CM en las líneas del Basic o en las variables del mismo Basic.

Anteriormente apuntabámos la conveniencia de inspeccionar el listado Basic del cargador ubicando el mismo en otra dirección a fin de no modificar en nada su contenido, esta necesidad es imperiosa en el caso de que el programa Basic tuviera CM enmascarado en el mismo.

Supongamos que existe una línea Basic en el medio del listado en el que, después de un REM, se halla un programa en CM; supongamos igualmente que todo el resto del listado se haya protegido con controles de color. Si quitamos éstos, el programa en CM se reubicará con lo que cuando el cargador lo llame el mismo no funcionara. De ahí la necesidad de ver el listado con el Copyline publicado en anteriores semanas.

El CM en líneas REM se reconoce por la visión de ésta y a continuación una serie de tokens y literales incoherentes. Cuando veas esto... NO LO TOQUES!!!, es mejor inspeccionarlo tranquilamente con un desensamblador. Modifica su dirección con el Copyline y examina su contenido.

EJEMPLO DE UTILIZACION DEL PROGRAMA COPYUPI EN LA MODIFICACION DE UNA CABECERA

```
COPYUPI        © 1985 MICROHOBBY
1 TIPO              program
2 NOMBRE            LOADER
3 LONGUITUD         107
4 COMIENZO          1
5 VARIABLES         107
6 TIPO DE FLAG  0
V - volver al menu
C - cambiar datos
```

Antes de modificar: Basic con autoejecución en línea 1

```
COPYUPI        © 1985 MICROHOBBY
1 TIPO              bytes
2 NOMBRE            LOADER
3 LONGUITUD         107
4 COMIENZO          30000
5 VARIABLES         107
6 TIPO DE FLAG  0
V - volver al menu
C - cambiar datos
```

Después de modificar: Bytes, ubicado a partir de la dirección 30000

CONTROLES DE COLOR Y CURSOR

VALOR	COMENTARIO
6	Control de print con coma, sirve para que en este punto el listado se desplace a la próxima posición de TAB. Va solo.
8	Cursor izquierda. Provoca el desplazamiento del cursor una posición a la izquierda sobreescribiéndose lo que a continuación vaya encima del anterior carácter.
9	Cursor derecha. Igual que el anterior sólo que hacia la derecha.
13	Código de Enter. Indica el final de una línea. Colocado en cualquier posición de una línea puede confundir al SO.
14	Código de un número. Precede a los cinco octetos que representan a un número en coma flotante.
16	Control de tinta. El código que le siga indicará de qué color se va a poner la tinta.
17	Control de papel. De igual manera que el control de tinta indica qué color va a tomar el papel a partir de este punto.
18	Control de flash. Indica si el flash está activado, si el próximo octeto es un 1, o no lo está, si el próximo octeto es un 0.
19	Control de brillo. Funciona de idéntica forma al control de flash.
20	Control de inverse. Como el control de flash y brillo.
21	Control de over. Como los tres anteriores.
22	Control de AT. Los dos octetos que le sigan indican las nuevas coordenadas por las que va a continuar el listado.
23	Control de TAB. Funciona igual que el control de AT, pero con un solo octeto que indica la nueva columna hacia la que se va a dirigir el listado.

CONSULTORIO

"Draco"

En la línea 12 del programa «Draco» publicado en el número 59 de su revista, debe haber algún error, ya que al ejecutar el programa aparece el mensaje «3 Subscript wrong 12:1». Les ruego me indiquen donde puede estar dicho error.

Aprovecho la oportunidad para hacerles dos preguntas más:

1. Si piensan ustedes publicar en cinta los programas aparecidos en esta segunda época de su revista.

2. Si tienen intención de comercializar algún tipo de fichero para guardar las fichas que publican con el Curso de Código Máquina.

Andrés DIAZ - Madrid

☐ La línea 12 del programa «Draco», se encarga de pasar a decimal los datos de las líneas 30 a 50 que están en hexa. El mensaje de error indica que el subíndice supera la longitud de la viariable «a$». Existen dos posibilidades: o bien la variable es demasiado corta, o bien el subíndice llega demasiado lejos. Empecemos por la segunda: «a$» deberá tener siempre 26 caracteres, por tanto, la variable «i» deberá ir desde 1 hasta 25 (para que «i + 1» no dé error); compruebe el quinto comando de la línea 10 que deberá ser:

FOR i = 1 TO 25 STEP 2

La otra posibilidad es que alguna de las líneas de datos tenga menos de 26 caracteres; puede comprobarlo tecleando:

PRINT L

cuando le salga el error: el resultado será el número de línea que tiene menos de 26 caracteres; compruébelo haciendo:

PRINT LEN a$

Respecto a sus otras preguntas, la respuesta es afirmativa para ambas. Cada cuatro números publicamos una cinta y cada 20 un cartón con cinco cintas. Por otro lado, estamos preparando el fichero que usted indica; ya avisaremos cuando esté disponible.

Lectura de teclado

Me gustaría saber la ubicación en la ROM de la rutina que es capaz de almacenar la variable del Sistema «LAST-K».

También quisiera saber la señal o señales que he de codificar para poder «latchear» el port 223 en su salida. Tengo un interface triestado para este port, pero lógicamente, la señal permanece un período de tiempo muy corto cuando realizo un «OUT».

Cristóbal LOPEZ - Málaga

☐ El teclado se lee mediante interrupción enmascarable en MODO 1, por lo que la variable «LAST-K» se actualiza, de forma automática, cada 20 milisegundos siempre que haya tecla pulsada; en caso contrario, conserva su anterior contenido. Si va a hacer una rutina que cambie el vector de interrupción, puede utilizar «RST #38» para leer el teclado y actualizar el contador de tiempo real. Si no quiere actualizar el contador («FRAMES»), puede hacer una llamada a la rutina «KEYBOARD» que se encarga de leer el teclado y actualizar «LAST-K». «KEYBOARD» está en la dirección 02BFh (703) y deberá preservar «AF», «HL», «BC» y «DE» antes de llamarla y recuperarlos después.

Respecto a su segunda pregunta, no estamos seguros de haber entendido bien su problema. Si dice que tiene un «interface» para ese port, suponemos que este se encargará de decodificar las señales IORQ y WR así como el bit A5 del bus de direcciones. Lo que tal vez necesite sea un «latch» a la salida de ese «interface» que tiene, para que almacene el dato entregado por el bus de datos en el momento de «abrirse» el port.

"INPUT" numérico

Sea el siguiente programa:

```
10 INPUT "Dame un número"; LINE a$
20 FOR I = 1 TO LEN a$:
   IF CODE a$ (I) > 57 OR CODE a$ (I) < 48 THEN GO TO 20
30 LET a = VAL a$
```

Bien, hasta aquí, el programa funciona no aceptando nada más que valores numéricos. Ahora bien, ¿de qué forma puedo evitar que se produzca error si en vez de darle un número o una letra, pulso «ENTER».

Otra pregunta: Si en un programa hacemos «CLEAR 32500», con lo cual hemos dejado menos memoria al Basic. cuando el ordenador me diga que no tiene más memoria, ¿hay alguna manera de volver a subrir la RAMTOP?

Francisco GARCIA - Málaga

☐ De momento, el «GO TO» de la línea 20 debe ser «GO TO 10» ya que, de lo contrario, entraríamos en un bucle sin fin. Además de eso, puede añadir la línea:

15 IF LEN a$ = 0 THEN GO TO 10 que detecta si la variable «a$» es una cadena vacía por haberse pulsado «ENTER» directamente.

Respecto a su segunda pregunta, puede volver a subir la RAMTOP haciendo «CLEAR» a una dirección más alta, pero el Basic le «manchacará» lo que tuviera por encima de 32500.

"ELITE"

Hace unos días me compré el juego «ELITE» de Firebird. Es muy bueno, pero tengo algunos problemas:

1. En las instrucciones dice que cuando se pulse la tecla «J», haremos un salto a «TORUS»; ¿Qué significa esto?

2. En la página 7 de las instrucciones en inglés, viene un título que dice: «HOW TO ACESS THE CASSETTE.» No viene traducida esta página al castellano; ayer intenté averiguar en qué consistia y después de jugar un rato, pulsé «2» (Save Commander); me pidió el nombre del comandante, luego me dijo que pusiera en marcha el cassette, y me apareció un número bastante largo en la pantalla. La grabación fue bastante corta. ¿Qué me grabó? ¿Qué significa el número que apareció?

José GUDE - Madrid

☐ 1. La tecla «J» sirve para que, cuando estemos en el modo navegación, procedamos a conseguir la máxima velocidad que puede coger la nave en un sistema planetario. Asimismo, existe un salto interplanetario que se consigue con la tecla «A» no sin antes haber fijado el destino en la

carta de los alrededores de la nave; y un salto intergaláctico que sirve para saltar a la próxima galaxia de entre las 8 que componen el universo; esto último lo conseguiremos con las teclas «H» y «J» pulsadas a la vez, eso sí, siempre y cuando hayamos comprado el salto en un planeta con nivel tecnológico superior a 10. Su precio es de 5.000 créditos.

2. Este juego es de los que te puedes tirar jugando toda la vida, por lo que tiene la posibilidad, estando aterrizado en una base, de grabar la partida como la llevemos, en una cinta para, en una posterior sesión, continuar. El código es para dar fe de los logros que hayamos conseguido y evitar que el usuario con virtudes «hackerianas» modifique los datos que se graban para llegar más lejos sin necesidad de jugar.

"SPEEDINK"

¿Se puede alterar la velocidad del comando «FLASH» para que su efecto se produzca más rápidamente?

¿Si se envía un programa a su revista para que lo editen, podría mandarse a otras para que lo editaran también?

Manuel GUTIERREZ - Málaga

☐ En algunos ordenadores es posible alterar la velocidad de parpadeo, por ejemplo, en el Amstrad se puede hacer con el comando «SPEEDINK». Desgraciadamente, no es el caso del Spectrum, aunque puede simularlo actuando directamente sobre los atributos del carácter o caracteres cuya velocidad de parpadeo quiera alterar.

Respecto a su segunda pregunta, evidentemente, puede hacerlo; pero no sería una práctica muy ética y, desde luego, perdería todas las posibilidades de publicar con nosotros (probablemente, a las otras revistas tampoco les gustaría demasiado). De todas formas y para no perjudicar a nuestros lectores, solemos comunicar a la mayor brevedad posible si el programa ha sido aceptado o no.

Fragmentación de cadenas

Desearía saber si existe alguna función en el Spectrum que sustituye a las funciones MID$, LEFT$, o RIGHT$ o si hay alguna manera de simularlas.

Iñaki PILDAIN - Pamplona

☐ Realmente no existe una función sustitutoria de éstas, sino una notación alternativa para fragmentar cadenas. El Spectrum fragmenta las cadenas como si se tratasen de vectores (arrays) de caracteres utilizando la notación (primero TO último). Para que se aclare, le pondremos un ejemplo. La sentencia:

LET b$ = LEFT$ (a$,3)

se podría simular en el Spectrum como:

LET b$ = a$ (1 TO 3)

Otro ejemplo. La sentencia:

LET b$ = RIGHT$ (a$,5)

podría ser:

LET b$ = a$(LEN a$—5 TO LEN a$)

o simplemente:

LET b$ = a$(LEN a$—5 TO)

ya que el inicio y final de cadena se asumen por defecto. En el caso de la función MID$ es aún más sencillo:

LET b$ = a MID$ (a$ 5, 12)

podría ser:

LET b$ = a$ (5 TO 12)

Está permitida la asignación de la cadena sobre sí misma con lo que está queda redimensionada (evidentemente, hacia menos; nunca hacia más). Con el tiempo, comprobará que este sistema es enormemente más ventajoso que el utilizar las funciones MID$, RIGHT$ y LEFT$.

Curiosidades del Código Máquina

Tengo el libro «C/M ZX Spectrum» de Indescomp y en él veo que el nemónico LD (nn), HL se repite dos veces en la lista de instrucciones. La primera se ensambla con 4 bytes (ED,63,n,n) y la segunda con sólo 3 (22,n,n). Mi pregunta es obvia: ¿Por qué dos instrucciones iguales? ¿Qué diferencia hay entre una y otra? He de deciros que mi ensamblador las ensambla con 4 bytes.

Manuel CIPRIAN - Barcelona

☐ Como ya hemos explicado en nuestro Curso de Código Máquina, el procedimiento que utiliza el Z-80 para decodificar las instrucciones cuyo código de operación tiene dos bytes, hace que, en ocasiones, se repitan instrucciones que ya existen entre las que se codifican con un código de operación de 1 byte. Esto ocurre, sobre todo, con las instrucciones que afectan a los registros «A» y «HL» por ser los principales del microprocesador. No hay que olvidar que el Z-80 se diseñó como una versión mejorada del famoso 8080 de Intel al que se le añadieron unas cuantas instrucciones que son, en su mayoría, las que se codifican con código de operación de dos bytes.

Evidentemente, los señores que diseñaron el Z-80 podrían haber evitado esto; pero hubiera supuesto complicar en exceso la circuitería de decodificación. Por otro lado, cualquier ensamblador que se precie debe elegir siempre la forma de codificación que ocupe menos bytes. Ignoramos qué ensamblador utiliza usted; pero, evidentemente, no debe ser muy bueno cuando comete esos fallos.

Joystick Kempston

He notado que cuando conecto el interface de joystick tipo Kempston con el Interface 1 conectado, el ordenador no responde a las órdenes del joystick. Si conecto éste sin el Interface 1 funciona perfectamente.

Juan GARCIA - Barcelona

☐ En numerosas ocasiones, hemos indicado que el joystick tipo Kempston debe leerse con la función: «IN 223». No obstante, hay programas comerciales que utilizan «IN 1» o «IN 31» para leerlo; esta forma funciona bien si no se tiene conectado el Interface 1, pero crea confusión si se utiliza este dispositivo. Por tanto, reiteramos que la forma correcta de leer el joystick tipo Kempston es con *«IN 223»* que funciona tanto con Interface 1 como sin él.

DE OCASION

● DESEARIA contactar con usuarios del Zx Spectrum 48K, que estén interesados en programar. Interesados pueden escribir a la siguiente dirección: Miguel Angel Aguilar Prenes. C/ Simón Jiménez Reina, BL-2. 3.° B. Málaga. Tel. 27 77 88 (mañanas).

● DESEARIA que algún lector me proporcionase fotocopia de las instrucciones del Compilador Pascal HP 4S de Hisoft. También busco copia Cinta Pruebas Zx TP de Sinclair. Pagaré gastos de envío o bien lo cambiaré por algo equivalente. Interesados dirigirse a Juan B. Guillén. C/ Mayor de Sarriá, 123. Barcelona. Tel. (93) 203 16 10.

● VENDO Spectrum 48K, más cassette, teclado profesional, 2 joystick Quick Shot II y Kempston interface Ram Turbo. Regalo 5 libros. Precio por todo el lote 45.000 ptas. (negociables). Interesados escribir a David Cebrián Piera. C/ Cosmógrafo Ramírez, 13. Xátiva (Valencia).

● VENDO ordenador Spectrum 48K, con cables, transformador, cinta Horizontes, manuales en inglés y castellano. Precio: 30.000 ptas. También vendo fotocopiadora DRY SUPERPONG por 3.000. Interesados llamar de 9 a 2 al tel. 218 73 68; o bien escribir a Carlos Sánchez. Crta. Boadilla del Monte, 25. 28024 Madrid.

● VENDO Spectrum 48K (3 meses de garantía). Regalo joystick Quick Shot II con interface Kempston, regalo revistas: 12 ZX, 3 Micromanía, 35 de Microhobby. Sólo por 32.000 ptas. (precio total 53.875). Manuel Díaz Fernández. Avda. de Portugal, 39. 32002 Orense.

● VENDO ordenador Zx Spectrum 48K con cables, fuente de alimentación, cassette especial, interface, joystick, 2 manuales del Spectrum, 1 libro de programación Basic, 50 revistas. Todo por 35.000 ptas. Interesados contactar con Fernando Civera. Tel. (976) 23 97 12.

● COMPRO Gens-3 o lo intercambio por otro. Interesados escribir a Fernando Avilés. Grupo Escolar s/n. Fuentealbilla (Albacete). Tel. (967) 46 08 34.

● VENDO Spectrum Plus, en perfecto estado. Precio: 32.000 ptas. Interesados llamar al tel. (93) 204 30 22. Ildefonso Lacasta. C/ Manila, 51. 08034 Barcelona.

● VENDO Spectrum 48K con garantía, fuente de alimentación, cables y manuales en castellano, al precio de 35.000 ptas. Regalo revistas. Dirigirse por carta o bien por tel. (94) 440 38 88. Félix Val. C/ Zumalacárregui, 1, 7.° C. Basauri (Vizcaya).

● VENDO Spectrum 48K por 27.000 ptas. (Precio a negociar). Regalo 20 revistas. J. Pablo Vázquez López. C/ Coruña, 45. Monforte (Lugo). Tel. (982) 40 22 24.

● AMIGOS DEL SPECTRUM 2 un nuevo club para el Spectrum. Solicitamos nuevos socios. Tenemos una revista mensual. Para más información llamar al tel. (965) 39 17 63, o bien dirigirse a Fco. José Román Asensi. C/ Gral. Monasterio, 43, 1.° B. Elda (Alicante).

● VENDO Zx Spectrum con ampliación interna a 48K. Libro de instrucciones, fuente de alimentación, cables y cinta de Horizontes. También vendo joystick Quick Shot, interface Indescomp programable. Regalo tres números de Microhobby Cassette. Todo por 34.500 ptas. Envío contra reembolso. Interesados pueden escribir a Jorge Pérez Rebolleda. C/ Corpus Christi, 4, 1.° B. 47005 Valladolid.

● VENDO ordenador Láser - Color Computer 310, con 16K, libro de programación Basic, especial Láser-310, con dos tomos de instrucciones de estudios, fuente de alimentación y cables por sólo 15.000 ptas. Interesados llamar al tel. (954) 74 21 52.

● VENDO Scalextric GP-51 por 8.000 ptas. Además, junto con los coches, pistas, peraltes adicionales, etc. Escribir a Miguel A. Manrique. C/ Ruiz Zorrilla, 18, 9° F. 39009 Santander (Cantabria).

● Cambio Zx Spectrum 48K comprado hace 7 meses. Lo cambio junto con el cassette, y revistas además del manual, por un ordenador Amstrad o MSX que estén en buen estado. Interesados llamar al tel. 692 43 24 o bien escribir a Monturiol, 62, ent. 4.°. Ripollet (Barcelona).

● VENDO Atari completamente nuevo, con 3 mandos, información, gastos de envío gratis. Alexis Gutiérrez. C/ Gutierra Rada, 2. Laredo (Cantabria). Precio 18.000 ptas. Tel. (942) 60 62 25.

● DESEARIA que algún lector me proporcionase las instrucciones en castellano del copiador Lerm MM2. Enviar a Diego Giner. Apdo. 232. Melilla.

● VENDO Spectrum 16K con garantía, cables, transformador, libros y revistas por sólo 15.000 ptas. Interesados llamar al tel. (91) 250 79 81. Preguntar por Ricardo (tardes).

● VENDO Zx Spectrum 48K con fuente de alimentación y todos los cables de conexión, cinta Horizontes, libros en castellano, cassette especial ordenador, amplificador de sonido, joystick, interface y maletín para colocar el ordenador y sus componentes. Precio: 48.500. Regalo revistas. Interesados escribir a Luis Nogués. C/ Padre Gil, 3. Reus (Tarragona). Tel. (977) 32 09 79.

● VENDO ordenador Zx Spectrum Plus, por 22.000 ptas. Interface 1 y Microdrive por el precio de 9.000 ptas. cada uno. Regalo revistas. Todo en perfecto estado. Interesados escribir a Iñigo de Pineda. Avda. Sarriá, 38. 08029 Barcelona. Tel. 321 93 48.

● VENDO Spectrum Plus 48K, más cassette, libro de instrucciones. Precio 25.000 ptas. El ordenador tiene todo su equipo: fuente de alimentación, cable, televisor y tomas de grabación. La antigüedad es de 3 meses y con muy poco uso. Interesados escribir a la siguiente dirección: Julio Ramón Morcilla. C/ Real, s/n. Lashes (Asturias). Tel. 85 01 26.

● VENDO Spectrum 48K, cables, transformador, manuales en español, cinta Horizontes, libro «Spectrum, cómo sacar el máximo rendimiento», interface programable joystick, revistas de Microhobby desde el n.° 1 y otras especializadas. Precio 50.000 ptas. Ignacio Arias Ubillos. Valladolid. Tel. (983) 33 23 73. Llamar a partir de las 10,30 de la noche.

● VENDO a precio muy interesante y en buen estado los siguientes números de la revista Zx: del 1 al 27; de Todospectrum 1, 2, y 4 a 16; de Tu Micro: 1 a 12; Micromanía: 1 y 2. Dos tapas para encuadernar 12 revistas de Zx o Todospectrum. Para mayor información dirigirse a Xavier Lamiquiz Pierron. Parque Simón Verde, 99. San Juan de Aznalfarache (Sevilla).

● VENDO interface programable, sin estrenar. Precio a convenir. Llamar al tel. (93) 726 37 73. Preguntar por Mateu (tardes).

● VENDO Spectrum 48K. Precio a convenir. Interesados escribir a la siguiente dirección: Ignacio Puertas Robina. C/ Fernández y González, 16-18, 2.° A. 41001 Sevilla o bien llamar al tel. (954) 22 26 10.

● CAMBIO vídeo-juego Soundic, con dos mandos para un teclado de Zx Spectrum Plus, en buenas condiciones o cualquier otro profesional. Interesados llamar al tel. (93) 340 09 78. Preguntar por Carlos (tardes).

● URGE vender Spectrum 16K ampliado a 48K con pocas horas de uso, en buen estado, con cables, transformador, libro de instrucciones en castellano. Incluyo libro y alguna revista. Precio a convenir. Interesados escribir a Avelino Martínez. C/ Alfredo Brañas, 10, 4.° C. Santiago de Compostela (La Coruña). Llamar al tel. (981) 59 67 29.


ZOIDS
LA BATALLA COMIENZA
ZAFI CHIP
Zoidstar está atrapado por la guerra. Zoid azul pelea con Zoid rojo en una despiadada batalla por la supremacía. El poderoso Zoidzilla, el zoid azul, líder del combate, mantiene dormidas sus habilidades, preparado para subir a la cápsula comando y fusionar su mente con el poder de la máquina para tomar el control de esta implacable confrontación. ¡En algún lugar de este desolado paisaje los Zoid Rojos acechan! Los preparativos para el desembarco están completos. El descenso empieza. La tarea que sigue te llevará a los límites de tu capacidad e imaginación.
¡VIVE EL JUEGO!
¡TRANSFORMATE EN MAQUINA!
SPECTRUM 48k COMMODORE 64 AMSTRAD MSX
Si deseas información y participar en los importantes sorteos que ZAFICHIP celebrará durante el año... ¡ESCRIBENOS!
Editado, fabricado y distribuido en España bajo la garantía Zafiro. Todos los derechos reservados.
ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141. 28046 Madrid.
Tel. 459 30 04. Tel. Barna. 209 33 65. Telex: 22690 ZAFIR E